

# BONUS

## 16 PAGE POCKET PROGRAMS









```

10020  UTAB 141: HTAB 141: PRINT "BY N.J. SMITH"
10030  GOSUB 20000
10040  UTAB 5
10050  PRINT "THERE ARE SIX BLOCKS, ALL OF DIFFERENT SIZES. THE PUZZLE ST
PARTS OFF WITH THE BLOCKS ARRANGED IN A TOWER"
10060  PRINT "THE TOWER IS POSITIONED ON ONE OF THREE SITES. YOU M
UST TRANSFER THE TOWER FROM ITS ORIGINAL SPOT TO ONE OF THE OTHER "T
"
10070  PRINT "PRINT "YOU MAY ONLY MOVE ONE BLOCK AT A TIME, AND YOU CAN
OT MOVE ONE THAT IS LYING UNDER ANOTHER"
10080  PRINT "PRINT "AT NO TIME CAN YOU MOVE A BLOCK ONTO A BLOCK OF SHO
LLER SIZE THAN ITSELF"
10090  GOSUB 20000
10100  UTAB 5
10110  PRINT "THE BLOCK YOU ARE CURRENTLY USING WILL BE ORANGE. TO CHANGE
BLOCKS, YOU MAY PRESS EITHER A OR C"
10120  PRINT
10130  PRINT "HEY A WILL SELECT A BLOCK THAT IS ONE SIZE SMALLER"
10140  PRINT
10150  PRINT "HEY C WILL SELECT A BLOCK THAT IS ONE SIZE LARGER"
10160  PRINT "END"
10170  PRINT "TO MOVE A BLOCK PRESS ONE OF THE ARROWS"
10180  PRINT
10190  GOSUB 20000
10200  UTAB 231: HTAB 65: PRINT "HIT SPACE BAR TO CONTINUE": IF PEEK C =
16384: GOTO 10210: THEN GOTO 10220
10210  POKE C = 16384: GOTO 10220
10220  RETURN

```

## Machine Code For BASIC

By Derek Au

THIS PROGRAM decompiles a machine-code file into an Applesoft program consisting of POKEs and line numbers. This is useful for having machine-language sub-routines ready poked into memory from BASIC.

```

10  PRINT "MOMON C,I,O"
20  HOME
30  PRINT
40  INPUT "ENTER STARTING ADDRESS?:";AS:
   A = VAL (AS)
50  PRINT : PRINT
60  INPUT "ENTER ENDING ADDRESS?:";XS:
   X = VAL (XS)
80  DIM LINE(2)
90  INPUT "YOUR DESIRED STARTING LINE?:";D
100 HOME
110 POKE 34,2: PRINT "SELECT MACHINE-CODE
FILE "
120 PRINT "CATALOG"
130 INPUT "FILENAME:->";FILES
140 PRINT "BLOAD";FILES
150 POKE 34,0
160 HOME
170 PRINT "OPEN CODE"
180 PRINT "DELETE CODE"
190 PRINT "OPEN CODE"
200 PRINT "WRITE CODE"
210 PRINT "DEL0,400": FOR PR = A TO VAL (XS)
220 CO = CO + 1
230 IF CO = 10 THEN CO = 1
240 IF CO < > 1 THEN 280
250 PRINT
260 PRINT D: " ";
270 D = D + 10
280 PRINT "POKE ";PR; ", "; PEEK (PR); " ";
290 NEXT PR
300 PRINT : PRINT "CLOSE CODE"
310 PRINT "MON C,I,O"
320 PRINT "EXEC CODE"
330 END

```

## Ullo Again...

By S. Zanker

WITH THIS program, you will know how much random-access memory (RAM) your computer contains, and which slots have disk cards, without having to open the cover.

You will also know how much memory is available to an Applesoft BASIC program, and where that program will be located, and the number of buffers available for disk files (current value of MAXFILES).

The program variables contain this information:

DAY\$ = Date of creation (you supply)  
 TIS\$ = Disk title (you supply)  
 MEM = Highest RAM address + 1  
 TY = Type of disk (SLAVE/MASTER)  
 TY\$ = Type of disk ('SLAVE'/MASTER')  
 HI = Current value of HIMEM  
 PR = Pointer to start of basic program

FR = Free memory for basic program  
 VOL = Volume number of disk  
 MF = Current value of MAXFILES  
 SP\$ = Five spaces for editing

Line 20 gives you the option of recording the date the disk was initialised. Line 30 enables you to give the disk a title. There is a 40-character limit.

Line 50 disables the INIT command, preventing loss of non-write-protected disks.

Line 60 displays the amount of RAM measured in 1024 bytes.

Lines 70 and 80 display the type of disk last booted (SLAVE/MASTER), and the creation date.

Line 90 prints an inverse bar on the screen.

Lines 110 to 130 give a formatted display of the variables concerning BASIC program memory.

Lines 140 and 150 poke and execute the machine-language routine which looks for disk cards.

Another inverse line is printed at line 160. The screen is now divided into three sections: the top section deals with system data, the middle section with free memory and the lower section with disk information, which is displayed by lines 170 and 180.

Line 190 asks if you want a CATALOG. Your response will default to yes.

Line 200 issues the command, if you want a second greeting program, replace the statement with PRINT CHR\$(4) RUN filename'.

Line 210 rings a bell and wipes the program from memory, ready for use.

The program can be run on any size system. You'll be surprised how much RAM you have remaining after zero page, the input buffer, screen memory and DOS have all had their share.

```

10  REM GREETING PROGRAM
20  DAYS = "01JUL82"
30  TITLES = "<> SUPPLY YOUR TITLE HERE <>"
40  MEM = ( PEEK (978) + 35 ) * 256:TY = ( PEEK
(978) + 25 ) * 256 + 254:TT = (41 -
LEN (TIS)) / 2:TT = TT + (TT < 1)
50  POKE MEM - 6833,96
60  TEXT : HOME : NORMAL : PRINT TAB (14);"APPLE
| | + "MEM / 1024;"K"
70  PRINT "DOS 3.3"; TAB (30);TY$ = "MASTER":
IF PEEK (TY) < > 54 THEN TY$ = "SLAVE": FLASH
80  PRINT TY$;"DISK"; NORMAL : PRINT TAB (
13);"CREATED ";DAYS
90  PRINT "INVERSE: PRINT TAB (40);" ": NORMAL
100 HI = PEEK (115) + PEEK (116) * 256:PR =
PEEK (103) + PEEK (104) * 256:FR = HI - PR +
1:VOL = PEEK (MEM - 2058):MF = PEEK (MEM -
5545):SP$ = " "
110 PRINT "HIMEM = "; RIGHTS (SP$ + STR$
(HI),5)
120 PRINT "PROGRAM = "; RIGHTS (SP$ + STR$
(PR),5); TAB (28);"MAXFILES = ";MF: PRINT TAB (
10);"-----"
130 PRINT "FRE MEM = "; RIGHTS (SP$ + STR$
(FR),5);" ("; INT (FR / 1024);"K)": PRINT
FOR A = 768 TO 810: READ B: POKE A,B: NEXT :
DATA 162,0,169,200,134,6,133,7,160,7,198,7,
165,7,201,192,240,24,177,6,217,1,251,208,239,
136,136,16,245,165,7,203,192,32,227,253,32,
72,249,76,8,3,96
150 PRINT "DISK CARDS ARE IN SLOT(S)": CALL
768: PRINT : PRINT
160 INVERSE : PRINT TAB (40);" ": NORMAL
170 PRINT TAB (13)"DISK VOLUME #";VOL: PRINT
180 PRINT TAB (TT);TIS$: PRINT : PRINT
190 PRINT "CATALOG [Y] [N] ?[Y]": HTAB PEEK
(36) - 1: POKE 49168,0: GET ANS: PRINT ANS
200 IF ANS < > "N" THEN PRINT CHR$
(4)"CATALOG"
210 CALL - 198: NEW

```

## Epson Connections

By John Marquet

I HAVE an Apple II fitted out as a Pascal program development tool. The system has 48 kilobytes of memory, two disks, a 20-line by 80-character monitor and an Epson MS80F/T printer. Using this system, I have developed a program for printing one or more copies of a UCSD Pascal text file on an Epson printer, with an optional character density or size.

The UCSD Pascal text editor can handle and format simple correspondence as well as sourcing programs. When the dense-print option is selected on the printer, a double-s is given on each character, and the matrix print becomes clearer.

The program uses some of the features of the UCSD Pascal - for example, the file-naming conventions.

```

program typetext;
var hardcop,infile: text;
    numbercopy,indent,loop: integer;
    newk, innam: string;
    contr: char;

procedure initialise;
begin
    writing;
    writeln('John's text output');
    writeln('Note that an input line with a leading "-" will');
    writeln('result in a page throw on output. ');
    innam:= ' ';
    write('Type name of input text file: ');
    readln(innam);
    if pos('-',innam)=0 then innam:=concat(innam,'.TEXT');
    (ss):
    reset(infile,innam);
    if ioread=0 then begin
        writeln('File not found');
        exit (program)
    end;
    (ss):
    contr:= ' ';
    writeln('Enter < for small, > for large print');
    writeln('else # for high density normal size print. ');
    writeln('Any other character gives normal size and density. ');
    write('Type of print? ');
    readln(contr);
    if contr = '<' then begin
        write('Offset (no. of char spaces in left margin) ?');
        readln(indent);
        end else indent:=20;
    write('No. of copies? ');
    readln(numbercopy);
    if numbercopy<0 then exit (program);
    writeln('Set top of form on printer, then key "return"');
    readln(newk);
    rewrite(hardcop,'printer');

end;
begin
    begin
        while true do begin
            initialise;
            repeat begin
                reset(infile);
                while (not eof(infile)) do begin
                    readln(infile,newk);
                    writeln(newk);
                    if pos('-',newk)=1 then
                        begin
                            write(hardcop,chr(12));
                            newk[1]:=' ';
                            end;
                    begin
                        case contr of
                            '<': begin
                                write(hardcop,chr(15));
                                for loop:=1 to indent do write(hardcop,' ');
                                end;
                            '>': write(hardcop,chr(14));
                            '#': write (hardcop,chr(27),'E');
                            end;
                    writeln(hardcop,newk);
                    case contr of
                            '<': write(hardcop,chr(18));
                            '>': write(hardcop,chr(27),'F');
                            end;
                    end;
                    numbercopy:=numbercopy-1;
                    if numbercopy>0 then write(hardcop,chr(12));
                end;
                until numbercopy=0;
            close(infile);
            close(hardcop);
        end;
    end.

```

## Hackatext

By Derek Au

HACKATEXT is a program which creates sequential files but allows you to input commas and colons which the MAKE TEXT program supplied on the System Master disk cannot - for example, BSAVE PROGRAM,A\$4000,L8192,S6,V100,D2.



```

10 REM MAKE TEXT
20 HOME
30 HTAB 10: PRINT "TEXT-FILE CREATOR#1"
40 HTAB 5: VTAB 7
50 PRINT " (M)AKE TEXT      ":
   PRINT " (C)ATALOG        ":
   PRINT " (E)XIT           ":
60 PRINT " WHICH?": GET B$
70 IF B$ = "M" THEN 110
80 IF B$ = "C" THEN PRINT: INPUT "PRESS
'RETURN' TO CATALOG;NS: PRINT CHR$ (4)"CATALOG":
INPUT "":NS: GOTO 20
90 IF B$ = "E" THEN HOME: PRINT "BYE!!": END
100 PRINT CHR$ (7): GOTO 20
110 PRINT: PRINT "MAKING TEXT: USE <CTRL-F> TO
ESCAPE.....<CTRL-R>RETURN..."
120 A = 8192
130 GET WS: PRINT WS;
140 IF WS = CHR$ (6) THEN 180
150 IF WS = CHR$ (18) THEN A = 8192: GOTO 20
160 POKE A, ASC (WS):A = A + 1
170 GOTO 130
180 INPUT "FINISHED?->":NS
190 IF LEFT$ (NS,1) = "Y" THEN 210
200 GOTO 130
210 INPUT "FILENAME?->":FILES
220 INPUT "(T)EXT-FILE OR (B)INARY-FILE?->":MS:
IF MS = "B" THEN 320
230 IF MS = "T" THEN 250
240 PRINT CHR$ (7): VTAB PEEK (37): GOTO 220
250 PRINT CHR$ (4)"OPEN":FILES
260 PRINT CHR$ (4)"WRITE":FILES
270 FOR WR = 8192 TO A
280 PRINT CHR$ ( PEEK (WR));
290 NEXT
300 PRINT: PRINT CHR$ (4)"CLOSE":FILES
310 GOTO 20
320 PRINT CHR$ (4)"BSAVE":FILES;"A8192,L";
A = 8192
330 GOTO 20

```

## Siren

By Robert Chalmers

HAVE YOU ever needed a short program that will sound an alarm for you at a pre-determined point in a main program?

Siren is a little machine-language program that will do just that. It is listed by code, hex and decimal, with a short demonstration program to show its use from within a larger framework. The starting address is 804 (dec), usually free for machine-code work.

```

10 FOR I = 804 TO 819
20 READ A
30 POKE I,A
40 NEXT I
50 DATA 160,9,169,106,153,208,7,32,58,255,200,192,40,208,243,96

```

To instigate this program from BASIC, use this method:

```

LDY #0          LOAD Y WITH 0
LDA #A          LOAD ACCUMULATOR WITH FLASHING *
STASB700,Y     STORE ACCUMULATOR AT $B700 + CONTENTS OF Y.
                (BOTTOM LINE OF SCREEN.)
JSR FFB3A      JUMP TO SUBROUTINE IN MONITOR THAT RINGS OR
                BELL
INY            INCREMENT Y REGISTER BY 1
CPY #256       COMPARE VALUE IN Y TO $256 (40 DEC)
BNE F3         BRANCH BACK 13 STEPS IF Y IS NOT EQUAL TO
                $256 (40)
RTS            RETURN TO MAIN PROGRAM

```

To demonstrate this in use, add the following:

```

60 PRINT "ENTER FIVE NUMBERS."
70 N=0:INPUT "INPUT A NUMBER.":B
80 LET N=N+1
90 PRINT B:IF N=5 THEN CALL 804
100 GOTO 70

```

## Merry Christmas

By Derek Au

THIS PROGRAM plays the traditional "We Wish You A Merry Christmas" tune. Just some short programming fun! The program POKES some sound routines which can be used in your own programs. Just: POKE 768,(pitch 1-255) : POKE 769,(duration 1-255) : CALL 770.

```

10 GOSUB 110
20 HOME
30 AS = "WE WISH YOU A MERRY CHRISTMAS",
  WE WISH YOU A MERRY CHRISTMAS', WE WISH
  YOU A MERRY CHRISTMAS', ... AND A 'HAPPY
  NEW YEAR!' "
40 DATA 133,110, 98,99, 98,99,88,80, 98,100,105,110,
  117,130, 117,150, 117,150, 88,110,
  88,110, 78,75, 88,99, 98,99, 105,150, 105,150,
  105,160, 78,100, 78,100, 67,110
50 DATA 78,100,88,100,98,100, 98,100, 133,130,
  133,130, 117,115, 88,120, 105,120, 98,200
60 FOR R = 1 TO 30
70 READ P,D
80 POKE 768,P: POKE 769,D: CALL 770
90 NEXT
100 END
110 POKE 768,0: POKE 769,173: POKE 770,173: POKE
  771,48: POKE 772,192: POKE 773,136: POKE 774,208:
  POKE 775,5: POKE 776,206:
120 POKE 777,1: POKE 778,3: POKE 779,240: POKE
  780,9: POKE 781,202: POKE 782,208: POKE 783,245:
  POKE 784,174: POKE 785,0:
130 POKE 786,3: POKE 787,76: POKE 788,2: POKE
  789,3: POKE 790,96: POKE 791,0:
140 RETURN

```

## OUR COURIERS DELIVER TO YOUR DOOR

### ORDER HOTLINE (02) 232-7704

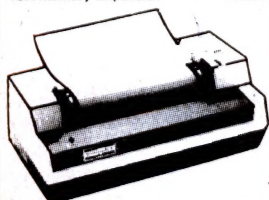
- ★ **BANKCARD**
- ★ **MOST ORDERS SHIPPED WITHIN 48 HRS**
- ★ **14-DAY MONEY BACK GUARANTEE**
- ★ **WE WILL TRY TO BEAT ANY CURRENTLY ADVERTISED PRICE**
- ★ **WRITE TO: G.P.O. BOX 4475, SYDNEY, N.S.W. 2001.**

## COMPUTER DISCOUNTERS



### MAIL ORDER ADVERTISING PRICE LIST

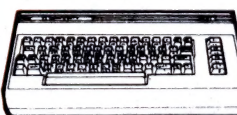
Vic 20 - Priced too low to list  
 Vic 1541 Single Disk Drive 599.00  
 Vic Datasette 84.00  
 Joystick Controller 16.00  
 Vic 1515 Graphics Printer 399.00  
 Vic Super Expander 59.00  
 3K Memory Expansion 45.00  
 8K Memory Expansion 69.00  
 16K Memory Expansion 110.00



**Software for VIC 20**  
 Games Pack 1 (3 programs) 16.95  
 Games Pack 2 (3 programs) 19.95  
 Ludwig's Lemon Laser 16.95  
 Kongo Kong 19.95  
 Annihilator 19.95  
 Trek 16.95  
 Bug Blast 19.95  
 Headon 17.95  
 Paratrooper 19.95  
 Cricket 16.95  
 Artillery 19.95  
 Alien Invasion 19.95  
 Adventurer Pack (8k or 16k) 19.95  
 Allied Defense 19.95  
 Mail it 20 19.95  
 Micro Hex Assembler Editor (3k or 8k) 19.95  
 Cosmic Crystals (paddles) 19.95  
 Rebel Defender (8k & paddles) 19.95  
 Cosmic Crusader 19.95  
 Blasteroids 16.95  
 Ultimate Tank 3k 19.95  
 Champ Chase 16.95  
 Galactic Crossfire 19.95  
 Cheese Book (3k, 8k or 16k) 24.95  
 Horse File (3k, 8k or 16k) 24.95  
 Grave Robbers 19.95  
 Dungeon Quest (16k) 19.95  
 Adventure Pack 1 (3 programs) 19.95  
 Adventure Pack 2 (3 programs) 19.95  
 4 in one Games Special 19.95

All Commodore VIC 20 games cartridges 35.00  
 Creative Software cartridge for VIC 20  
 Choplifter, Trashman, Serpentine, Astra Blitz, Apple Panic, All 49.95  
 3 slot memory expansion board now with on/off and reset switch 59.95  
 Light pens now available for VIC 20  
 CALL FOR PRICE

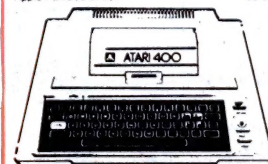
### COMMODORE 64 CALL FOR PRICE



**COMMODORE 64 SOFTWARE**  
 Grave Robbers 19.95  
 Annihilator 24.95  
 Trek 16.95  
 Adventurer Pack 1 (3 programs) 19.95  
 Adventurer Pack 2 (3 programs) 19.95

**HARDWARE**  
 Atari 800 Computer 48K with Basic 1225.00  
 Atari 400 Computer 48K 639.00  
 Atari 410 Program Recorder 139.00  
 Atari 810 Disk Drive 769.00  
 Atari Joystick per Pair 36.00

**SOFTWARE**  
 Wordprocessor (Disk) 289.00  
 Personal Financial Management (D) 149.00  
 Conversational French (T) 79.00  
 Conversational German (T) 79.00  
 Conversational Spanish (T) 79.00  
 Conversational Italian (T) 79.00  
 Pacman (Cart) 65.00  
 Centipede (C) 65.00  
 Star Raiders (C) 65.00  
 Space Invaders (C) 53.00  
 Missile Command (C) 53.00  
 Asteroids (C) 53.00  
 Apple Panic (C or D) 39.95



## KAYPRO

**KAYPRO II COMPUTER** Including SBASIC CPM, Profit Plan, Perfect Calc, Perfect Writer, Perfect Speller and Perfect Filer 64K Memory, 8" Monitor, 200K per disk, fully portable. PHONE FOR OUR VERY SPECIAL PRICE!!!



### PRINTERS

Brother HRI DaisyWheel 15" 1299.00  
 CITOH 8510 80 Col Dot Matrix 995.00  
 CITOH1550 132 Col DotMatrix 1195.00  
 CITOH F1040 Daisy Wheel 15" 2095.00  
 Star Printer Centronics Parallel 529.00

### FREIGHT CHARGES

Under 5KG - ADD \$5.00  
 5-10KG - ADD \$7.50  
 PRICES SUBJECT TO CHANGE

```

0000  *****MODIFIED VIDEO DRIVER*****
0010
0020      ORG 401EH
0030      DEFW DRIVER          ;NEW JUMP VECTOR
0040
0050      ORG 7F00H
0060  CURSOR DEFB 08H          ;CURSOR CHARACTER
0070  LINESW DEFB 08H          ;LINES NOT TO SCROLL
0080  DRIVER  LD  L,(IX+03)
0090          LD  H,(IX+04)      ;HL=CURSOR POSITION
0100          JP  C,0404H
0110          LD  A,(IX+05)
0120          OR  A             ;AS COVERED CHARACTER
0130          JR  Z,0FF1H
0140          LD  (HL),A
0150  OFF1  LD  A,C
0160          CP  0             ;GO IF TABS OR GRAPHICS
0170          JR  C,CNTRL
0180          OR  SOR            ;GO IF UPPER CASE
0190          JR  NC,GRAPHIC
0200          CP  08H
0210          JR  C,DIS
0220          LD  B,A
0230          LD  A,(HL),A
0240          LD  A,(HL)
0250          CP  D
0260          JP  Z,DIS
0270          CP  08H
0280          SUB  08H
0290  DIS  CALL DISPLY

```



[illegible][illegible]

## Modified CP/M

By Nigel Harwood

ON BOOT-UP of my Osborne 1 computer, it automatically executes a program called AUTOST.COM, which enables any program or function to be automatically started.

This is very useful, but not always desirable. However, when I decided that it wasn't wanted for a particular disk, I ran into some problems. You see, nowhere in the manual does it mention *how* to disengage the AUTOST facility.

I tried simply deleting the program AUTOST.COM, but then CP/M booted up with the message "AUTOST?" ... which, to me, looked very poor.

The answer I found was in slightly changing the CP/M itself. Execute SYSGEN, to the source-drive prompt, enter A. To the destination-drive prompt, enter RETURN. This will then have read in the CP/M system and left it in memory. Now, enter SAVE 35 CPM.COM, which will save the first 35 pages of memory; these contain the CP/M system.

Next, execute DDT CPM.COM, then in DDT enter D2008. You should now see:

```
2008 E5 01 07 41 55 54 4F 53...AUTOS
2010 54 20 30 31 32 33 34 35 T 012345
```

Now, enter S2008 and then the following: RETURN, RETURN, 00,20,20,20,20,20, ending by entering a fullstop and new line. Next, enter Control-C to get back to the CP/M operating system.

You have now modified the CP/M system - it was read into memory when you entered DDT, and it is still there now.

Execute SYSGEN again, answer RETL to the source-drive prompt, because you wish to use the CP/M that you have modified in memory. To the destination-drive prompt, enter A.

When the disk light has gone out, press the reset button to reboot the system. After hitting RETURN to the boot-up prompt, the improved sign-on should greet you on a cleared screen.



## Mastermind

By Tony Hinde

YOU USE the standard rules of the Mastermind board game for this program. No colours can be repeated, and the code at the end is set out at random, not in order.

```
10 CLS:CLER1000
20 RANDOM
40 PRINT:THIS GAME IS THE COMPUTERISED VERSION OF MASTER MIND
70 PRINT:I HOPE YOU ENJOY IT.
100 PRINT:PRINT HIT ANY KEY TO CONTINUE.....
130 IFPEEK(13359) THEN 130
140 PRINT:YOU HAVE A SELECTION OF EIGHT COLOURS.
150 PRINT:R FOR RED, Y FOR YELLOW, G FOR GREEN, B FOR BLUE
160 PRINT:O FOR ORANGE, W FOR WHITE, P FOR PINK AND S FOR SILVER.
170 PRINT:YOU MUST GUESS THE COMBINATION OF FIVE COLOURS THAT
180 PRINT:THE COMPUTER HAS SELECTED. YOU HAVE 10 ATTEMPTS.
190 PRINT:CHUNK143 MEANS RIGHT COLOUR RIGHT SOURCE.
200 PRINT:CHUNK143 MEANS RIGHT COLOUR WRONG SOURCE.
210 PRINT:CHUNK134 MEANS RIGHT COLOUR WRONG SOURCE.
220 CLS
230 PRINT:CHUNK134 MEANS RIGHT COLOUR WRONG SOURCE.
240 PRINT:
250 PRINT:TO CORRECT THE COMBINATION YOUR WRITING PRESS THE
260 PRINT:BACKSPACE KEY.
270 PRINT:PRINT HIT ANY KEY TO PLAY.....
400 IFPEEK(13359) THEN 400
410 FOR I=1 TO 5
420 FOR J=1 TO 8
430 FOR K=1 TO 8
440 FOR L=1 TO 8
450 FOR M=1 TO 8
460 FOR N=1 TO 8
470 FOR O=1 TO 8
480 FOR P=1 TO 8
490 FOR Q=1 TO 8
500 FOR R=1 TO 8
510 FOR S=1 TO 8
520 FOR T=1 TO 8
530 FOR U=1 TO 8
540 FOR V=1 TO 8
550 FOR W=1 TO 8
560 FOR X=1 TO 8
570 FOR Y=1 TO 8
580 FOR Z=1 TO 8
590 FOR AA=1 TO 8
600 FOR AB=1 TO 8
610 FOR AC=1 TO 8
620 FOR AD=1 TO 8
630 FOR AE=1 TO 8
640 FOR AF=1 TO 8
650 FOR AG=1 TO 8
660 FOR AH=1 TO 8
670 FOR AI=1 TO 8
680 FOR AJ=1 TO 8
690 FOR AK=1 TO 8
700 FOR AL=1 TO 8
710 FOR AM=1 TO 8
720 FOR AN=1 TO 8
730 FOR AO=1 TO 8
740 FOR AP=1 TO 8
750 FOR AQ=1 TO 8
760 FOR AR=1 TO 8
770 FOR AS=1 TO 8
780 FOR AT=1 TO 8
790 FOR AU=1 TO 8
800 FOR AV=1 TO 8
810 FOR AW=1 TO 8
820 FOR AX=1 TO 8
830 FOR AY=1 TO 8
840 FOR AZ=1 TO 8
850 FOR BA=1 TO 8
860 FOR BB=1 TO 8
870 FOR BC=1 TO 8
880 FOR BD=1 TO 8
890 FOR BE=1 TO 8
900 FOR BF=1 TO 8
910 FOR BG=1 TO 8
920 FOR BH=1 TO 8
930 FOR BI=1 TO 8
940 FOR BJ=1 TO 8
950 FOR BK=1 TO 8
960 FOR BL=1 TO 8
970 FOR BM=1 TO 8
980 FOR BN=1 TO 8
990 FOR BO=1 TO 8
1000 FOR BP=1 TO 8
1010 FOR BQ=1 TO 8
1020 FOR BR=1 TO 8
1030 FOR BS=1 TO 8
1040 FOR BT=1 TO 8
1050 FOR BU=1 TO 8
1060 FOR BV=1 TO 8
1070 FOR BW=1 TO 8
1080 FOR BX=1 TO 8
1090 FOR BY=1 TO 8
1100 FOR BZ=1 TO 8
1110 FOR CA=1 TO 8
1120 FOR CB=1 TO 8
1130 FOR CC=1 TO 8
1140 FOR CD=1 TO 8
1150 FOR CE=1 TO 8
1160 FOR CF=1 TO 8
1170 FOR CG=1 TO 8
1180 FOR CH=1 TO 8
1190 FOR CI=1 TO 8
1200 FOR CJ=1 TO 8
1210 FOR CK=1 TO 8
1220 FOR CL=1 TO 8
1230 FOR CM=1 TO 8
1240 FOR CN=1 TO 8
1250 FOR CO=1 TO 8
1260 FOR CP=1 TO 8
1270 FOR CQ=1 TO 8
1280 FOR CR=1 TO 8
1290 FOR CS=1 TO 8
1300 FOR CT=1 TO 8
1310 FOR CU=1 TO 8
1320 FOR CV=1 TO 8
1330 FOR CW=1 TO 8
1340 FOR CX=1 TO 8
1350 FOR CY=1 TO 8
1360 FOR CZ=1 TO 8
1370 FOR DA=1 TO 8
1380 FOR DB=1 TO 8
1390 FOR DC=1 TO 8
1400 FOR DD=1 TO 8
1410 FOR DE=1 TO 8
1420 FOR DF=1 TO 8
1430 FOR DG=1 TO 8
1440 FOR DH=1 TO 8
1450 FOR DI=1 TO 8
1460 FOR DJ=1 TO 8
1470 FOR DK=1 TO 8
1480 FOR DL=1 TO 8
1490 FOR DM=1 TO 8
1500 FOR DN=1 TO 8
1510 FOR DO=1 TO 8
1520 FOR DP=1 TO 8
1530 FOR DQ=1 TO 8
1540 FOR DR=1 TO 8
1550 FOR DS=1 TO 8
1560 FOR DT=1 TO 8
1570 FOR DU=1 TO 8
1580 FOR DV=1 TO 8
1590 FOR DW=1 TO 8
1600 FOR DX=1 TO 8
1610 FOR DY=1 TO 8
1620 FOR DZ=1 TO 8
1630 FOR EA=1 TO 8
1640 FOR EB=1 TO 8
1650 FOR EC=1 TO 8
1660 FOR ED=1 TO 8
1670 FOR EE=1 TO 8
1680 FOR EF=1 TO 8
1690 FOR EG=1 TO 8
1700 FOR EH=1 TO 8
1710 FOR EI=1 TO 8
1720 FOR EJ=1 TO 8
1730 FOR EK=1 TO 8
1740 FOR EL=1 TO 8
1750 FOR EM=1 TO 8
1760 FOR EN=1 TO 8
1770 FOR EO=1 TO 8
1780 FOR EP=1 TO 8
1790 FOR EQ=1 TO 8
1800 FOR ER=1 TO 8
1810 FOR ES=1 TO 8
1820 FOR ET=1 TO 8
1830 FOR EU=1 TO 8
1840 FOR EV=1 TO 8
1850 FOR EW=1 TO 8
1860 FOR EX=1 TO 8
1870 FOR EY=1 TO 8
1880 FOR EZ=1 TO 8
1890 FOR FA=1 TO 8
1900 FOR FB=1 TO 8
1910 FOR FC=1 TO 8
1920 FOR FD=1 TO 8
1930 FOR FE=1 TO 8
1940 FOR FF=1 TO 8
1950 FOR FG=1 TO 8
1960 FOR FH=1 TO 8
1970 FOR FI=1 TO 8
1980 FOR FJ=1 TO 8
1990 FOR FK=1 TO 8
2000 FOR FL=1 TO 8
2010 FOR FM=1 TO 8
2020 FOR FN=1 TO 8
2030 FOR FO=1 TO 8
2040 FOR FP=1 TO 8
2050 FOR FQ=1 TO 8
2060 FOR FR=1 TO 8
2070 FOR FS=1 TO 8
2080 FOR FT=1 TO 8
2090 FOR FU=1 TO 8
2100 FOR FV=1 TO 8
2110 FOR FW=1 TO 8
2120 FOR FX=1 TO 8
2130 FOR FY=1 TO 8
2140 FOR FZ=1 TO 8
2150 FOR GA=1 TO 8
2160 FOR GB=1 TO 8
2170 FOR GC=1 TO 8
2180 FOR GD=1 TO 8
2190 FOR GE=1 TO 8
2200 FOR GF=1 TO 8
2210 FOR GH=1 TO 8
2220 FOR GI=1 TO 8
2230 FOR GJ=1 TO 8
2240 FOR GK=1 TO 8
2250 FOR GL=1 TO 8
2260 FOR GM=1 TO 8
2270 FOR GN=1 TO 8
2280 FOR GO=1 TO 8
2290 FOR GP=1 TO 8
2300 FOR GQ=1 TO 8
2310 FOR GR=1 TO 8
2320 FOR GS=1 TO 8
2330 FOR GT=1 TO 8
2340 FOR GU=1 TO 8
2350 FOR GV=1 TO 8
2360 FOR GW=1 TO 8
2370 FOR GX=1 TO 8
2380 FOR GY=1 TO 8
2390 FOR GZ=1 TO 8
2400 FOR HA=1 TO 8
2410 FOR HB=1 TO 8
2420 FOR HC=1 TO 8
2430 FOR HD=1 TO 8
2440 FOR HE=1 TO 8
2450 FOR HF=1 TO 8
2460 FOR HG=1 TO 8
2470 FOR HH=1 TO 8
2480 FOR HI=1 TO 8
2490 FOR HJ=1 TO 8
2500 FOR HK=1 TO 8
2510 FOR HL=1 TO 8
2520 FOR HM=1 TO 8
2530 FOR HN=1 TO 8
2540 FOR HO=1 TO 8
2550 FOR HP=1 TO 8
2560 FOR HQ=1 TO 8
2570 FOR HR=1 TO 8
2580 FOR HS=1 TO 8
2590 FOR HT=1 TO 8
2600 FOR HU=1 TO 8
2610 FOR HV=1 TO 8
2620 FOR HW=1 TO 8
2630 FOR HX=1 TO 8
2640 FOR HY=1 TO 8
2650 FOR HZ=1 TO 8
2660 FOR IA=1 TO 8
2670 FOR IB=1 TO 8
2680 FOR IC=1 TO 8
2690 FOR ID=1 TO 8
2700 FOR IE=1 TO 8
2710 FOR IF=1 TO 8
2720 FOR IG=1 TO 8
2730 FOR IH=1 TO 8
2740 FOR II=1 TO 8
2750 FOR IJ=1 TO 8
2760 FOR IK=1 TO 8
2770 FOR IL=1 TO 8
2780 FOR IM=1 TO 8
2790 FOR IN=1 TO 8
2800 FOR IO=1 TO 8
2810 FOR IP=1 TO 8
2820 FOR IQ=1 TO 8
2830 FOR IR=1 TO 8
2840 FOR IS=1 TO 8
2850 FOR IT=1 TO 8
2860 FOR IU=1 TO 8
2870 FOR IV=1 TO 8
2880 FOR IW=1 TO 8
2890 FOR IX=1 TO 8
2900 FOR IY=1 TO 8
2910 FOR IZ=1 TO 8
2920 FOR JA=1 TO 8
2930 FOR JB=1 TO 8
2940 FOR JC=1 TO 8
2950 FOR JD=1 TO 8
2960 FOR JE=1 TO 8
2970 FOR JF=1 TO 8
2980 FOR JG=1 TO 8
2990 FOR JH=1 TO 8
3000 FOR JI=1 TO 8
3010 FOR JJ=1 TO 8
3020 FOR JK=1 TO 8
3030 FOR JL=1 TO 8
3040 FOR JM=1 TO 8
3050 FOR JN=1 TO 8
3060 FOR JO=1 TO 8
3070 FOR JP=1 TO 8
3080 FOR JQ=1 TO 8
3090 FOR JR=1 TO 8
3100 FOR JS=1 TO 8
3110 FOR JT=1 TO 8
3120 FOR JU=1 TO 8
3130 FOR JV=1 TO 8
3140 FOR JW=1 TO 8
3150 FOR JX=1 TO 8
3160 FOR JY=1 TO 8
3170 FOR JZ=1 TO 8
3180 FOR KA=1 TO 8
3190 FOR KB=1 TO 8
3200 FOR KC=1 TO 8
3210 FOR KD=1 TO 8
3220 FOR KE=1 TO 8
3230 FOR KF=1 TO 8
3240 FOR KG=1 TO 8
3250 FOR KH=1 TO 8
3260 FOR KI=1 TO 8
3270 FOR KJ=1 TO 8
3280 FOR KK=1 TO 8
3290 FOR KL=1 TO 8
3300 FOR KM=1 TO 8
3310 FOR KN=1 TO 8
3320 FOR KO=1 TO 8
3330 FOR KP=1 TO 8
3340 FOR KQ=1 TO 8
3350 FOR KR=1 TO 8
3360 FOR KS=1 TO 8
3370 FOR KT=1 TO 8
3380 FOR KU=1 TO 8
3390 FOR KV=1 TO 8
3400 FOR KW=1 TO 8
3410 FOR KX=1 TO 8
3420 FOR KY=1 TO 8
3430 FOR KZ=1 TO 8
3440 FOR LA=1 TO 8
3450 FOR LB=1 TO 8
3460 FOR LC=1 TO 8
3470 FOR LD=1 TO 8
3480 FOR LE=1 TO 8
3490 FOR LF=1 TO 8
3500 FOR LG=1 TO 8
3510 FOR LH=1 TO 8
3520 FOR LI=1 TO 8
3530 FOR LJ=1 TO 8
3540 FOR LK=1 TO 8
3550 FOR LL=1 TO 8
3560 FOR LM=1 TO 8
3570 FOR LN=1 TO 8
3580 FOR LO=1 TO 8
3590 FOR LP=1 TO 8
3600 FOR LQ=1 TO 8
3610 FOR LR=1 TO 8
3620 FOR LS=1 TO 8
3630 FOR LT=1 TO 8
3640 FOR LU=1 TO 8
3650 FOR LV=1 TO 8
3660 FOR LW=1 TO 8
3670 FOR LX=1 TO 8
3680 FOR LY=1 TO 8
3690 FOR LZ=1 TO 8
3700 FOR MA=1 TO 8
3710 FOR MB=1 TO 8
3720 FOR MC=1 TO 8
3730 FOR MD=1 TO 8
3740 FOR ME=1 TO 8
3750 FOR MF=1 TO 8
3760 FOR MG=1 TO 8
3770 FOR MH=1 TO 8
3780 FOR MI=1 TO 8
3790 FOR MJ=1 TO 8
3800 FOR MK=1 TO 8
3810 FOR ML=1 TO 8
3820 FOR MM=1 TO 8
3830 FOR MN=1 TO 8
3840 FOR MO=1 TO 8
3850 FOR MP=1 TO 8
3860 FOR MQ=1 TO 8
3870 FOR MR=1 TO 8
3880 FOR MS=1 TO 8
3890 FOR MT=1 TO 8
3900 FOR MU=1 TO 8
3910 FOR MV=1 TO 8
3920 FOR MW=1 TO 8
3930 FOR MX=1 TO 8
3940 FOR MY=1 TO 8
3950 FOR MZ=1 TO 8
3960 FOR NA=1 TO 8
3970 FOR NB=1 TO 8
3980 FOR NC=1 TO 8
3990 FOR ND=1 TO 8
4000 FOR NE=1 TO 8
4010 FOR NF=1 TO 8
4020 FOR NG=1 TO 8
4030 FOR NH=1 TO 8
4040 FOR NI=1 TO 8
4050 FOR NJ=1 TO 8
4060 FOR NK=1 TO 8
4070 FOR NL=1 TO 8
4080 FOR NM=1 TO 8
4090 FOR NO=1 TO 8
4100 FOR NP=1 TO 8
4110 FOR NQ=1 TO 8
4120 FOR NR=1 TO 8
4130 FOR NS=1 TO 8
4140 FOR NT=1 TO 8
4150 FOR NU=1 TO 8
4160 FOR NV=1 TO 8
4170 FOR NW=1 TO 8
4180 FOR NX=1 TO 8
4190 FOR NY=1 TO 8
4200 FOR NZ=1 TO 8
4210 FOR OA=1 TO 8
4220 FOR OB=1 TO 8
4230 FOR OC=1 TO 8
4240 FOR OD=1 TO 8
4250 FOR OE=1 TO 8
4260 FOR OF=1 TO 8
4270 FOR OG=1 TO 8
4280 FOR OH=1 TO 8
4290 FOR OI=1 TO 8
4300 FOR OJ=1 TO 8
4310 FOR OK=1 TO 8
4320 FOR OL=1 TO 8
4330 FOR OM=1 TO 8
4340 FOR ON=1 TO 8
4350 FOR OO=1 TO 8
4360 FOR OP=1 TO 8
4370 FOR OQ=1 TO 8
4380 FOR OR=1 TO 8
4390 FOR OS=1 TO 8
4400 FOR OT=1 TO 8
4410 FOR OU=1 TO 8
4420 FOR OV=1 TO 8
4430 FOR OW=1 TO 8
4440 FOR OX=1 TO 8
4450 FOR OY=1 TO 8
4460 FOR OZ=1 TO 8
4470 FOR PA=1 TO 8
4480 FOR PB=1 TO 8
4490 FOR PC=1 TO 8
4500 FOR PD=1 TO 8
4510 FOR PE=1 TO 8
4520 FOR PF=1 TO 8
4530 FOR PG=1 TO 8
4540 FOR PH=1 TO 8
4550 FOR PI=1 TO 8
4560 FOR PJ=1 TO 8
4570 FOR PK=1 TO 8
4580 FOR PL=1 TO 8
4590 FOR PM=1 TO 8
4600 FOR PN=1 TO 8
4610 FOR PO=1 TO 8
4620 FOR PP=1 TO 8
4630 FOR PQ=1 TO 8
4640 FOR PR=1 TO 8
4650 FOR PS=1 TO 8
4660 FOR PT=1 TO 8
4670 FOR PU=1 TO 8
4680 FOR PV=1 TO 8
4690 FOR PW=1 TO 8
4700 FOR PX=1 TO 8
4710 FOR PY=1 TO 8
4720 FOR PZ=1 TO 8
4730 FOR QA=1 TO 8
4740 FOR QB=1 TO 8
4750 FOR QC=1 TO 8
4760 FOR QD=1 TO 8
4770 FOR QE=1 TO 8
4780 FOR QF=1 TO 8
4790 FOR QG=1 TO 8
4800 FOR QH=1 TO 8
4810 FOR QI=1 TO 8
4820 FOR QJ=1 TO 8
4830 FOR QK=1 TO 8
4840 FOR QL=1 TO 8
4850 FOR QM=1 TO 8
4860 FOR QN=1 TO 8
4870 FOR QO=1 TO 8
4880 FOR QP=1 TO 8
4890 FOR QQ=1 TO 8
4900 FOR QR=1 TO 8
4910 FOR QS=1 TO 8
4920 FOR QT=1 TO 8
4930 FOR QU=1 TO 8
4940 FOR QV=1 TO 8
4950 FOR QW=1 TO 8
4960 FOR QX=1 TO 8
4970 FOR QY=1 TO 8
4980 FOR QZ=1 TO 8
4990 FOR RA=1 TO 8
5000 FOR RB=1 TO 8
5010 FOR RC=1 TO 8
5020 FOR RD=1 TO 8
5030 FOR RE=1 TO 8
5040 FOR RF=1 TO 8
5050 FOR RG=1 TO 8
5060 FOR RH=1 TO 8
5070 FOR RI=1 TO 8
5080 FOR RJ=1 TO 8
5090 FOR RK=1 TO 8
5100 FOR RL=1 TO 8
5110 FOR RM=1 TO 8
5120 FOR RN=1 TO 8
5130 FOR RO=1 TO 8
5140 FOR RP=1 TO 8
5150 FOR RQ=1 TO 8
5160 FOR RR=1 TO 8
5170 FOR RS=1 TO 8
5180 FOR RT=1 TO 8
5190 FOR RU=1 TO 8
5200 FOR RV=1 TO 8
5210 FOR RW=1 TO 8
5220 FOR RX=1 TO 8
5230 FOR RY=1 TO 8
5240 FOR RZ=1 TO 8
5250 FOR SA=1 TO 8
5260 FOR SB=1 TO 8
5270 FOR SC=1 TO 8
5280 FOR SD=1 TO 8
5290 FOR SE=1 TO 8
5300 FOR SF=1 TO 8
5310 FOR SG=1 TO 8
5320 FOR SH=1 TO 8
5330 FOR SI=1 TO 8
5340 FOR SJ=1 TO 8
5350 FOR SK=1 TO 8
5360 FOR SL=1 TO 8
5370 FOR SM=1 TO 8
5380 FOR SN=1 TO 8
5390 FOR SO=1 TO 8
5400 FOR SP=1 TO 8
5410 FOR SQ=1 TO 8
5420 FOR SR=1 TO 8
5430 FOR SS=1 TO 8
5440 FOR ST=1 TO 8
5450 FOR SU=1 TO 8
5460 FOR SV=1 TO 8
5470 FOR SW=1 TO 8
5480 FOR SX=1 TO 8
5490 FOR SY=1 TO 8
5500 FOR SZ=1 TO 8
5510 FOR TA=1 TO 8
5520 FOR TB=1 TO 8
5530 FOR TC=1 TO 8
5540 FOR TD=1 TO 8
5550 FOR TE=1 TO 8
5560 FOR TF=1 TO 8
5570 FOR TG=1 TO 8
5580 FOR TH=1 TO 8
5590 FOR TI=1 TO 8
5600 FOR TJ=1 TO 8
5610 FOR TK=1 TO 8
5620 FOR TL=1 TO 8
5630 FOR TM=1 TO 8
5640 FOR TN=1 TO 8
5650 FOR TO=1 TO 8
5660 FOR TP=1 TO 8
5670 FOR TQ=1 TO 8
5680 FOR TR=1 TO 8
5690 FOR TS=1 TO 8
5700 FOR TT=1 TO 8
5710 FOR TU=1 TO 8
5720 FOR TV=1 TO 8
5730 FOR TW=1 TO 8
5740 FOR TX=1 TO 8
5750 FOR TY=1 TO 8
5760 FOR TZ=1 TO 8
5770 FOR UA=1 TO 8
5780 FOR UB=1 TO 8
5790 FOR UC=1 TO 8
5800 FOR UD=1 TO 8
5810 FOR UE=1 TO 8
5820 FOR UF=1 TO 8
5830 FOR UG=1 TO 8
5840 FOR UH=1 TO 8
5850 FOR UI=1 TO 8
5860 FOR UJ=1 TO 8
5870 FOR UK=1 TO 8
5880 FOR UL=1 TO 8
5890 FOR UM=1 TO 8
5900 FOR UN=1 TO 8
5910 FOR UO=1 TO 8
5920 FOR UP=1 TO 8
5930 FOR UQ=1 TO 8
5940 FOR UR=1 TO 8
5950 FOR US=1 TO 8
5960 FOR UT=1 TO 8
5970 FOR UJ=1 TO 8
5980 FOR UV=1 TO 8
5990 FOR UW=1 TO 8
6000 FOR UX=1 TO 8
6010 FOR UY=1 TO 8
6020 FOR UZ=1 TO 8
6030 FOR VA=1 TO 8
6040 FOR VB=1 TO 8
6050 FOR VC=1 TO 8
6060 FOR VD=1 TO 8
6070 FOR VE=1 TO 8
6080 FOR VF=1 TO 8
6090 FOR VG=1 TO 8
6100 FOR VH=1 TO 8
6110 FOR VI=1 TO 8
6120 FOR VJ=1 TO 8
6130 FOR VK=1 TO 8
6140 FOR VL=1 TO 8
6150 FOR VM=1 TO 8
6160 FOR VN=1 TO 8
6170 FOR VO=1 TO 8
6180 FOR VP=1 TO 8
6190 FOR VQ=1 TO 8
6200 FOR VR=1 TO 8
6210 FOR VS=1 TO 8
6220 FOR VT=1 TO 8
6230 FOR VU=1 TO 8
6240 FOR VV=1 TO 8
6250 FOR VW=1 TO 8
6260 FOR VX=1 TO 8
6270 FOR VY=1 TO 8
6280 FOR VZ=1 TO 8
6290 FOR WA=1 TO 8
6300 FOR WB=1 TO 8
6310 FOR WC=1 TO 8
6320 FOR WD=1 TO 8
6330 FOR WE=1 TO 8
6340 FOR WF=1 TO 8
6350 FOR WG=1 TO 8
6360 FOR WH=1 TO 8
6370 FOR WI=1 TO 8
6380 FOR WJ=1 TO 8
6390 FOR WK=1 TO 8
6400 FOR WL=1 TO 8
6410 FOR WM=1 TO 8
6420 FOR WN=1 TO 8
6430 FOR WO=1 TO 8
6440 FOR WP=1 TO 8
6450 FOR WQ=1 TO 8
6460 FOR WR=1 TO 8
6470 FOR WS=1 TO 8
6480 FOR WT=1 TO 8
6490 FOR WU=1 TO 8
6500 FOR WV=1 TO 8
6510 FOR WW=1 TO 8
6520 FOR WX=1 TO 8
6530 FOR WY=1 TO 8
6540 FOR WZ=1 TO 8
6550 FOR XA=1 TO 8
6560 FOR XB=1 TO 8
6570 FOR XC=1 TO 8
6580 FOR XD=1 TO 8
6590 FOR XE=1 TO 8
6600 FOR XF=1 TO 8
6610 FOR XG=1 TO 8
6620 FOR XH=1 TO 8
6630 FOR XI=1 TO 8
6640 FOR XJ=1 TO 8
6650 FOR XK=1 TO 8
6660 FOR XL=1 TO 8
6670 FOR XM=1 TO 8
6680 FOR XN=1 TO 8
6690 FOR XO=1 TO 8
6700 FOR XP=1 TO 8
6710 FOR XQ=1 TO 8
6720 FOR XR=1 TO 8
6730 FOR XS=1 TO 8
6740 FOR XT=1 TO 8
6750 FOR XU=1 TO 8
6760 FOR XV=1 TO 8
6770 FOR XW=1 TO 8
6780 FOR XX=1 TO 8
6790 FOR XY=1 TO 8
6800 FOR XZ=1 TO 8
6810 FOR YA=1 TO 8
6820 FOR YB=1 TO 8
6830 FOR YC=1 TO 8
6840 FOR YD=1 TO 8
6850 FOR YE=1 TO 8
6860 FOR YF=1 TO 8
6870 FOR YG=1 TO 8
6880 FOR YH=1 TO 8
6890 FOR YI=1 TO 8
6900 FOR YJ=1 TO 8
6910 FOR YK=1 TO 8
6920 FOR YL=1 TO 8
6930 FOR YM=1 TO 8
6940 FOR YN=1 TO 8
6950 FOR YO=1 TO 8
6960 FOR YP=1 TO 8
6970 FOR YQ=1 TO 8
6980 FOR YR=1 TO 8
6990 FOR YS=1 TO 8
7000 FOR YT=1 TO 8
7010 FOR YU=1 TO 8
7020 FOR YV=1 TO 8
7030 FOR YW=1 TO 8
7040 FOR YX=1 TO 8
7050 FOR YY=1 TO 8
7060 FOR YZ=1 TO 8
7070 FOR ZA=1 TO 8
7080 FOR ZB=1 TO 8
7090 FOR ZC=1 TO 8
7100 FOR ZD=1 TO 8
7110 FOR ZE=1 TO 8
7120 FOR ZF=1 TO 8
7130 FOR ZG=1 TO 8
7140 FOR ZH=1 TO 8
7150 FOR ZI=1 TO 8
7160 FOR ZJ=1 TO 8
7170 FOR ZK=1 TO 8
7180 FOR ZL=1 TO 8
7190 FOR ZM=1 TO 8
7200 FOR ZN=1 TO 8
7210 FOR ZO=1 TO 8
7220 FOR ZP=1 TO 8
7230 FOR ZQ=1 TO 8
7240 FOR ZR=1 TO 8
7250 FOR ZS=1 TO 8
7260 FOR ZT=1 TO 8
7270 FOR ZU=1 TO 8
7280 FOR ZV=1 TO 8
7290 FOR ZW=1 TO 8
7300 FOR ZX=1 TO 8
7310 FOR ZY=1 TO 8
7320 FOR ZZ=1 TO 8
7330 FOR AA=1 TO 8
7340 FOR AB=1 TO 8
7350 FOR AC=1 TO 8
7360 FOR AD=1 TO 8
7370 FOR AE=1 TO 8
7380 FOR AF=1 TO 8
7390 FOR AG=1 TO 8
7400 FOR AH=1 TO 8
7410 FOR AI=1 TO 8
7420 FOR AJ=1 TO 8
7430 FOR AK=1 TO 8
7440 FOR AL=1 TO 8
7450 FOR AM=1 TO 8
7460 FOR AN=1 TO 8
7470 FOR AO=1 TO 8
7480 FOR AP=1 TO 8
7490 FOR AQ=1 TO 8
7500 FOR AR=1 TO 8
7510 FOR AS=1 TO 8
7520 FOR AT=1 TO 8
7530 FOR AU=1 TO 8
7540 FOR AV=1 TO 8
7550 FOR AW=1 TO 8
7560 FOR AX=1 TO 8
7570 FOR AY=1 TO 8
7580 FOR AZ=1 TO 8
7590 FOR BA=1 TO 8
7600 FOR BB=1 TO 8
7610 FOR BC=1 TO 8
7620 FOR BD=1 TO 8
7630 FOR BE=1 TO 8
7640 FOR BF=1 TO 8
7650 FOR BG=1 TO 8
7660 FOR BH=1 TO 8
7670 FOR BI=1 TO 8
7680 FOR BJ=1 TO 8
7690 FOR BK=1 TO 8
7700 FOR BL=1 TO 8
7710 FOR BM=1 TO 8
7720 FOR BN=1 TO 8
7730 FOR BO=1 TO 8
7740 FOR BP=1 TO 8
7750 FOR BQ=1 TO 8
7760 FOR BR=1 TO 8
7770 FOR BS=1 TO 8
7780 FOR BT=1 TO 8
7790 FOR BU=1 TO 8
7800 FOR BV=1 TO 8
7810 FOR BW=1 TO 8
7820 FOR BX=1 TO 8
7830 FOR BY=1 TO 8
7840 FOR BZ=1 TO 8
7850 FOR CA=1 TO 8
7860 FOR CB=1 TO 8
7870 FOR CC=1 TO 8
7880 FOR CD=1 TO 8
7890 FOR CE=1 TO 8
7900 FOR CF=1 TO 8
7910 FOR CG=1 TO 8
7920 FOR CH=1 TO 8
7930 FOR CI=1 TO 8
7940 FOR CJ=1 TO 8
7950 FOR CK=1 TO 8
7960 FOR CL=1 TO 8
7970 FOR CM=1 TO 8
7980 FOR CN=1 TO 8
7990 FOR CO=1 TO 8
8000 FOR CP=1 TO 8
8010 FOR CQ=1 TO 8
8020 FOR CR=1 TO 8
8030 FOR CS=1 TO 8
8040 FOR CT=1 TO 8
8050 FOR CU=1 TO 8
8060 FOR CV=1 TO 8
8070 FOR CW=1 TO 8
8080 FOR CX=1 TO 8
8090 FOR CY=1 TO 8
8100 FOR CZ=1 TO 8
8110 FOR DA=1 TO 8
8120 FOR DB=1 TO 8
8130 FOR DC=1 TO 8
8140 FOR DD=1 TO 8
8150 FOR DE=1 TO 8
8160 FOR DF=1 TO 8
8170 FOR DG=1 TO 8
8180 FOR DH=1 TO 8
8190 FOR DI=1 TO 8
8200 FOR DJ=1 TO 8
8210 FOR DK=1 TO 8
8220 FOR DL=1 TO 8
8230 FOR DM=1 TO 8
8240 FOR DN=1 TO 8
8250 FOR DO=1 TO 8
8260 FOR DP=1 TO 8
8270 FOR DQ=1 TO 8
8280 FOR DR=1 TO 8
8290 FOR DS=1 TO 8
8300 FOR DT=1 TO 8
8310 FOR DU=1 TO 8
8320 FOR DV=1 TO 8
8330 FOR DW=1 TO 8
8340 FOR DX=1 TO 8
8350 FOR DY=1 TO 8
8360 FOR DZ=1 TO 8
8370 FOR EA=1 TO 8
8380 FOR EB=1 TO 8
8390 FOR EC=1 TO 8
8400 FOR ED=1 TO 8
8410 FOR EE=1 TO 8
8420 FOR EF=1 TO 8
8430 FOR EG=1 TO 8
8440 FOR EH=1 TO 8
8450 FOR EI=1 TO 8
8460 FOR EJ=1 TO 8
8470 FOR EK=1 TO 8
8480 FOR EL=1 TO 8
8490 FOR EM=1 TO 8
8500 FOR EN=1 TO 8
8510 FOR EO=1 TO 8
8520 FOR EP=1 TO 8
8530 FOR EQ=1 TO 8
8540 FOR ER=1 TO 8
8550 FOR ES=1 TO 8
8560 FOR ET=1 TO 8
8570 FOR EU=1 TO 8
8580 FOR EV=1 TO 8
8590 FOR EW=1 TO 8
8600 FOR EX=1 TO 8
8610 FOR EY=1 TO 8
8620 FOR EZ=1 TO 8
8630 FOR FA=1 TO 8
8640 FOR FB=1 TO 8
8650 FOR FC=1 TO 8
8660 FOR FD=1 TO 8
8670 FOR FE=1 TO 8
8680 FOR FF=1 TO 8
8690 FOR FG=1 TO 8
8700 FOR FH=1 TO 8
8710 FOR FI=1 TO 8
8720 FOR FJ=1 TO 8
8730 FOR FK=1 TO 8
8740 FOR FL=1 TO 8
8750 FOR FM=1 TO 8
8760 FOR FN=1 TO 8
8770 FOR FO=1 TO 8
8780 FOR FP=1 TO 8
8790 FOR FQ=1 TO 8
8800 FOR FR=1 TO 8
8810 FOR FS=1 TO 8
8820 FOR FT=1 TO 8
8830 FOR FU=1 TO 8
8840 FOR FV=1 TO 8
8850 FOR FW=1 TO 8
8860 FOR FX=1 TO 8
8870 FOR FY=1 TO 8
8880 FOR FZ=1 TO 8
8890 FOR GA=1 TO 8
8900 FOR GB=1 TO 8
8910 FOR GC=1 TO 8
8920 FOR GD=1 TO 8
8930 FOR GE=1 TO 8
8940 FOR GF=1 TO 8
8950 FOR GG=1 TO 8
8960 FOR GH=1 TO 8
8970 FOR GI=1 TO 8
8980 FOR GJ=1 TO 8
8990 FOR GK=1 TO 8
9000 FOR GL=1 TO 8
9010 FOR GM=1 TO 8
9020 FOR GN=1 TO 8
9030 FOR GO=1 TO 8
9040 FOR GP=1 TO 8
9050 FOR GQ=1 TO 8
9060 FOR GR=1 TO 8
9070 FOR GS=1 TO 8
9080 FOR GT=1 TO 8
9090 FOR GU=1 TO 8
9100 FOR GV=1 TO 8
9110 FOR GW=1 TO 8
9120 FOR GX=1 TO 8
9130 FOR GY=1 TO 8
9140 FOR GZ=1 TO 8
9150 FOR HA=1 TO 8
9160 FOR HB=1 TO 8
9170 FOR HC=1 TO 8
9180 FOR HD=1 TO 8
9190 FOR HE=1 TO 8
9200 FOR HF=1 TO 8
9210 FOR HG=1 TO 8
9220 FOR HH=1 TO 8
9230 FOR HI=1 TO 8
9240 FOR HJ=1 TO 8
9250 FOR HK=1 TO 8
9260 FOR HL=1 TO 8
9270 FOR HM=1 TO 8
9280 FOR HN=1 TO 8
9290 FOR HO=1 TO 8
9300 FOR HP=1 TO 8
9310 FOR HQ=1 TO 8
9320 FOR HR=1 TO 8
9330 FOR HS=1 TO 8
9340 FOR HT
```



# MICRO DECISION



The Micro Decision is a professional, CP/M® based computer system that comes with the hardware and software needed for the large or small business. This system is ideal for the professional or small business owner who wants to enter the computer age at an amazingly low price. It offers all the word processing, financial planning and programming tools needed to increase productivity.

The big office with a backlog of work on its minis and mainframes can use the Micro Decision to generate budget projections, sales reports, resource utilization studies, off-book receivables . . . all those special projects that need to be done now, but don't justify tying up a big machine.

The Micro Decision is a single board, Z80®-based computer that comes with 64K of RAM, an on-board disk controller and one or two 5 1/4 inch floppy disk drives. An expansion port allows the Micro Decision to handle up to four 5 1/4 inch drives, while two RS-232 serial ports connect the Micro Decision to a terminal, and a printer or modem.

Standard software includes Wordstar®, the popular word processing program, and Correct-It®, a powerful spelling checker. Financial forecasting is easy with LogiCalc®, the electronic "spreadsheet" that adds computer power to business projections. Programmers will appreciate the choice of three programming languages: Microsoft® Basic-80, BaZic®, and Morrow Designs' Pilot. The software package is completed by the CP/M 2.2 Operating System, the industry standard O/S that allows access to more than 2,000 business programs.

Morrow Designs' optional terminal offers a full 12 inch diagonal, high resolution screen with 80 characters by 24 lines. The detachable keyboard allows positioning for maximum user comfort. An extra RS-232 serial port provides opportunities for additional I/O communication. And the terminals' intelligent features significantly improve system performance.

Add a printer and you have all the tools necessary to produce and store letters, documents and financial reports.

The Micro Decision is more than just the sum total of its hardware and software specifications. Morrow Designs has added features to the Micro Decision that make it convenient for the beginner as well as the experienced CP/M user. These features include: multiple diskette formats, enhanced error handling, Virtual Drive, simple terminal configuration, extensive diagnostics, a menu-driven front end, and perhaps most important, a clearly written user manual.

The Micro Decision's straightforward component design insures continued reliability. Each system is given extensive quality control and testing before shipment.

Morrow Designs has been dedicated to high-performance, low-cost computing since 1976. First in solid state memory, then disk memory, today in both single and multi-user computer systems.

Automation Statham Pty. Ltd. has been a Morrow designs dealer since 1976.

\*Plus Sales Tax if applicable

## AUTOMATION STATHAM PTY. LTD.

47 Birch St, Bankstown, NSW 2200. Australia. Phone (02) 709-4144.

In Brisbane: The Softwarehouse (07) 221-9640

In Melbourne: Microtrix (03) 439-5257

## Alarm Clock

By Philip Cookson

THE ALARM on this digital alarm clock can either be a "beep" alarm or a cassette on/off command, which can be programmed to record from radio or television.

```

100 *****
101 *****
102 *****
103 *****
104 *****
105 *****
106 *****
107 *****
108 *****
109 *****
110 *****
111 *****
112 *****
113 *****
114 *****
115 *****
116 *****
117 *****
118 *****
119 *****
120 *****
121 *****
122 *****
123 *****
124 *****
125 *****
126 *****
127 *****
128 *****
129 *****
130 *****
131 *****
132 *****
133 *****
134 *****
135 *****
136 *****
137 *****
138 *****
139 *****
140 *****
141 *****
142 *****
143 *****
144 *****
145 *****
146 *****
147 *****
148 *****
149 *****
150 *****
151 *****
152 *****
153 *****
154 *****
155 *****
156 *****
157 *****
158 *****
159 *****
160 *****
161 *****
162 *****
163 *****
164 *****
165 *****
166 *****
167 *****
168 *****
169 *****
170 *****
171 *****
172 *****
173 *****
174 *****
175 *****
176 *****
177 *****
178 *****
179 *****
180 *****
181 *****
182 *****
183 *****
184 *****
185 *****
186 *****
187 *****
188 *****
189 *****
190 *****
191 *****
192 *****
193 *****
194 *****
195 *****
196 *****
197 *****
198 *****
199 *****
200 *****
201 *****
202 *****
203 *****
204 *****
205 *****
206 *****
207 *****
208 *****
209 *****
210 *****
211 *****
212 *****
213 *****
214 *****
215 *****
216 *****
217 *****
218 *****
219 *****
220 *****
221 *****
222 *****
223 *****
224 *****
225 *****
226 *****
227 *****
228 *****
229 *****
230 *****
231 *****
232 *****
233 *****
234 *****
235 *****
236 *****
237 *****
238 *****
239 *****
240 *****
241 *****
242 *****
243 *****
244 *****
245 *****
246 *****
247 *****
248 *****
249 *****
250 *****
251 *****
252 *****
253 *****
254 *****
255 *****
256 *****
257 *****
258 *****
259 *****
260 *****
261 *****
262 *****
263 *****
264 *****
265 *****
266 *****
267 *****
268 *****
269 *****
270 *****
271 *****
272 *****
273 *****
274 *****
275 *****
276 *****
277 *****
278 *****
279 *****
280 *****
281 *****
282 *****
283 *****
284 *****
285 *****
286 *****
287 *****
288 *****
289 *****
290 *****
291 *****
292 *****
293 *****
294 *****
295 *****
296 *****
297 *****
298 *****
299 *****
300 *****
301 *****
302 *****
303 *****
304 *****
305 *****
306 *****
307 *****
308 *****
309 *****
310 *****
311 *****
312 *****
313 *****
314 *****
315 *****
316 *****
317 *****
318 *****
319 *****
320 *****
321 *****
322 *****
323 *****
324 *****
325 *****
326 *****
327 *****
328 *****
329 *****
330 *****
331 *****
332 *****
333 *****
334 *****
335 *****
336 *****
337 *****
338 *****
339 *****
340 *****
341 *****
342 *****
343 *****
344 *****
345 *****
346 *****
347 *****
348 *****
349 *****
350 *****
351 *****
352 *****
353 *****
354 *****
355 *****
356 *****
357 *****
358 *****
359 *****
360 *****
361 *****
362 *****
363 *****
364 *****
365 *****
366 *****
367 *****
368 *****
369 *****
370 *****
371 *****
372 *****
373 *****
374 *****
375 *****
376 *****
377 *****
378 *****
379 *****
380 *****
381 *****
382 *****
383 *****
384 *****
385 *****
386 *****
387 *****
388 *****
389 *****
390 *****
391 *****
392 *****
393 *****
394 *****
395 *****
396 *****
397 *****
398 *****
399 *****
400 *****
401 *****
402 *****
403 *****
404 *****
405 *****
406 *****
407 *****
408 *****
409 *****
410 *****
411 *****
412 *****
413 *****
414 *****
415 *****
416 *****
417 *****
418 *****
419 *****
420 *****
421 *****
422 *****
423 *****
424 *****
425 *****
426 *****
427 *****
428 *****
429 *****
430 *****
431 *****
432 *****
433 *****
434 *****
435 *****
436 *****
437 *****
438 *****
439 *****
440 *****
441 *****
442 *****
443 *****
444 *****
445 *****
446 *****
447 *****
448 *****
449 *****
450 *****
451 *****
452 *****
453 *****
454 *****
455 *****
456 *****
457 *****
458 *****
459 *****
460 *****
461 *****
462 *****
463 *****
464 *****
465 *****
466 *****
467 *****
468 *****
469 *****
470 *****
471 *****
472 *****
473 *****
474 *****
475 *****
476 *****
477 *****
478 *****
479 *****
480 *****
481 *****
482 *****
483 *****
484 *****
485 *****
486 *****
487 *****
488 *****
489 *****
490 *****
491 *****
492 *****
493 *****
494 *****
495 *****
496 *****
497 *****
498 *****
499 *****
500 *****
501 *****
502 *****
503 *****
504 *****
505 *****
506 *****
507 *****
508 *****
509 *****
510 *****
511 *****
512 *****
513 *****
514 *****
515 *****
516 *****
517 *****
518 *****
519 *****
520 *****
521 *****
522 *****
523 *****
524 *****
525 *****
526 *****
527 *****
528 *****
529 *****
530 *****
531 *****
532 *****
533 *****
534 *****
535 *****
536 *****
537 *****
538 *****
539 *****
540 *****
541 *****
542 *****
543 *****
544 *****
545 *****
546 *****
547 *****
548 *****
549 *****
550 *****
551 *****
552 *****
553 *****
554 *****
555 *****
556 *****
557 *****
558 *****
559 *****
560 *****
561 *****
562 *****
563 *****
564 *****
565 *****
566 *****
567 *****
568 *****
569 *****
570 *****
571 *****
572 *****
573 *****
574 *****
575 *****
576 *****
577 *****
578 *****
579 *****
580 *****
581 *****
582 *****
583 *****
584 *****
585 *****
586 *****
587 *****
588 *****
589 *****
590 *****
591 *****
592 *****
593 *****
594 *****
595 *****
596 *****
597 *****
598 *****
599 *****
600 *****
601 *****
602 *****
603 *****
604 *****
605 *****
606 *****
607 *****
608 *****
609 *****
610 *****
611 *****
612 *****
613 *****
614 *****
615 *****
616 *****
617 *****
618 *****
619 *****
620 *****
621 *****
622 *****
623 *****
624 *****
625 *****
626 *****
627 *****
628 *****
629 *****
630 *****
631 *****
632 *****
633 *****
634 *****
635 *****
636 *****
637 *****
638 *****
639 *****
640 *****
641 *****
642 *****
643 *****
644 *****
645 *****
646 *****
647 *****
648 *****
649 *****
650 *****
651 *****
652 *****
653 *****
654 *****
655 *****
656 *****
657 *****
658 *****
659 *****
660 *****
661 *****
662 *****
663 *****
664 *****
665 *****
666 *****
667 *****
668 *****
669 *****
670 *****
671 *****
672 *****
673 *****
674 *****
675 *****
676 *****
677 *****
678 *****
679 *****
680 *****
681 *****
682 *****
683 *****
684 *****
685 *****
686 *****
687 *****
688 *****
689 *****
690 *****
691 *****
692 *****
693 *****
694 *****
695 *****
696 *****
697 *****
698 *****
699 *****
700 *****
701 *****
702 *****
703 *****
704 *****
705 *****
706 *****
707 *****
708 *****
709 *****
710 *****
711 *****
712 *****
713 *****
714 *****
715 *****
716 *****
717 *****
718 *****
719 *****
720 *****
721 *****
722 *****
723 *****
724 *****
725 *****
726 *****
727 *****
728 *****
729 *****
730 *****
731 *****
732 *****
733 *****
734 *****
735 *****
736 *****
737 *****
738 *****
739 *****
740 *****
741 *****
742 *****
743 *****
744 *****
745 *****
746 *****
747 *****
748 *****
749 *****
750 *****
751 *****
752 *****
753 *****
754 *****
755 *****
756 *****
757 *****
758 *****
759 *****
760 *****
761 *****
762 *****
763 *****
764 *****
765 *****
766 *****
767 *****
768 *****
769 *****
770 *****
771 *****
772 *****
773 *****
774 *****
775 *****
776 *****
777 *****
778 *****
779 *****
780 *****
781 *****
782 *****
783 *****
784 *****
785 *****
786 *****
787 *****
788 *****
789 *****
790 *****
791 *****
792 *****
793 *****
794 *****
795 *****
796 *****
797 *****
798 *****
799 *****
800 *****
801 *****
802 *****
803 *****
804 *****
805 *****
806 *****
807 *****
808 *****
809 *****
810 *****
811 *****
812 *****
813 *****
814 *****
815 *****
816 *****
817 *****
818 *****
819 *****
820 *****
821 *****
822 *****
823 *****
824 *****
825 *****
826 *****
827 *****
828 *****
829 *****
830 *****
831 *****
832 *****
833 *****
834 *****
835 *****
836 *****
837 *****
838 *****
839 *****
840 *****
841 *****
842 *****
843 *****
844 *****
845 *****
846 *****
847 *****
848 *****
849 *****
850 *****
851 *****
852 *****
853 *****
854 *****
855 *****
856 *****
857 *****
858 *****
859 *****
860 *****
861 *****
862 *****
863 *****
864 *****
865 *****
866 *****
867 *****
868 *****
869 *****
870 *****
871 *****
872 *****
873 *****
874 *****
875 *****
876 *****
877 *****
878 *****
879 *****
880 *****
881 *****
882 *****
883 *****
884 *****
885 *****
886 *****
887 *****
888 *****
889 *****
890 *****
891 *****
892 *****
893 *****
894 *****
895 *****
896 *****
897 *****
898 *****
899 *****
900 *****
901 *****
902 *****
903 *****
904 *****
905 *****
906 *****
907 *****
908 *****
909 *****
910 *****
911 *****
912 *****
913 *****
914 *****
915 *****
916 *****
917 *****
918 *****
919 *****
920 *****
921 *****
922 *****
923 *****
924 *****
925 *****
926 *****
927 *****
928 *****
929 *****
930 *****
931 *****
932 *****
933 *****
934 *****
935 *****
936 *****
937 *****
938 *****
939 *****
940 *****
941 *****
942 *****
943 *****
944 *****
945 *****
946 *****
947 *****
948 *****
949 *****
950 *****
951 *****
952 *****
953 *****
954 *****
955 *****
956 *****
957 *****
958 *****
959 *****
960 *****
961 *****
962 *****
963 *****
964 *****
965 *****
966 *****
967 *****
968 *****
969 *****
970 *****
971 *****
972 *****
973 *****
974 *****
975 *****
976 *****
977 *****
978 *****
979 *****
980 *****
981 *****
982 *****
983 *****
984 *****
985 *****
986 *****
987 *****
988 *****
989 *****
990 *****
991 *****
992 *****
993 *****
994 *****
995 *****
996 *****
997 *****
998 *****
999 *****
1000 *****

```

```

360 *****
361 *****
362 *****
363 *****
364 *****
365 *****
366 *****
367 *****
368 *****
369 *****
370 *****
371 *****
372 *****
373 *****
374 *****
375 *****
376 *****
377 *****
378 *****
379 *****
380 *****
381 *****
382 *****
383 *****
384 *****
385 *****
386 *****
387 *****
388 *****
389 *****
390 *****
391 *****
392 *****
393 *****
394 *****
395 *****
396 *****
397 *****
398 *****
399 *****
400 *****
401 *****
402 *****
403 *****
404 *****
405 *****
406 *****
407 *****
408 *****
409 *****
410 *****
411 *****
412 *****
413 *****
414 *****
415 *****
416 *****
417 *****
418 *****
419 *****
420 *****
421 *****
422 *****
423 *****
424 *****
425 *****
426 *****
427 *****
428 *****
429 *****
430 *****
431 *****
432 *****
433 *****
434 *****
435 *****
436 *****
437 *****
438 *****
439 *****
440 *****
441 *****
442 *****
443 *****
444 *****
445 *****
446 *****
447 *****
448 *****
449 *****
450 *****
451 *****
452 *****
453 *****
454 *****
455 *****
456 *****
457 *****
458 *****
459 *****
460 *****
461 *****
462 *****
463 *****
464 *****
465 *****
466 *****
467 *****
468 *****
469 *****
470 *****
471 *****
472 *****
473 *****
474 *****
475 *****
476 *****
477 *****
478 *****
479 *****
480 *****
481 *****
482 *****
483 *****
484 *****
485 *****
486 *****
487 *****
488 *****
489 *****
490 *****
491 *****
492 *****
493 *****
494 *****
495 *****
496 *****
497 *****
498 *****
499 *****
500 *****
501 *****
502 *****
503 *****
504 *****
505 *****
506 *****
507 *****
508 *****
509 *****
510 *****
511 *****
512 *****
513 *****
514 *****
515 *****
516 *****
517 *****
518 *****
519 *****
520 *****
521 *****
522 *****
523 *****
524 *****
525 *****
526 *****
527 *****
528 *****
529 *****
530 *****
531 *****
532 *****
533 *****
534 *****
535 *****
536 *****
537 *****
538 *****
539 *****
540 *****
541 *****
542 *****
543 *****
544 *****
545 *****
546 *****
547 *****
548 *****
549 *****
550 *****
551 *****
552 *****
553 *****
554 *****
555 *****
556 *****
557 *****
558 *****
559 *****
560 *****
561 *****
562 *****
563 *****
564 *****
565 *****
566 *****
567 *****
568 *****
569 *****
570 *****
571 *****
572 *****
573 *****
574 *****
575 *****
576 *****
577 *****
578 *****
579 *****
580 *****
581 *****
582 *****
583 *****
584 *****
585 *****
586 *****
587 *****
588 *****
589 *****
590 *****
591 *****
592 *****
593 *****
594 *****
595 *****
596 *****
597 *****
598 *****
599 *****
600 *****
601 *****
602 *****
603 *****
604 *****
605 *****
606 *****
607 *****
608 *****
609 *****
610 *****
611 *****
612 *****
613 *****
614 *****
615 *****
616 *****
617 *****
618 *****
619 *****
620 *****
621 *****
622 *****
623 *****
624 *****
625 *****
626 *****
627 *****
628 *****
629 *****
630 *****
631 *****
632 *****
633 *****
634 *****
635 *****
636 *****
637 *****
638 *****
639 *****
640 *****
641 *****
642 *****
643 *****
644 *****
645 *****
646 *****
647 *****
648 *****
649 *****
650 *****
651 *****
652 *****
653 *****
654 *****
655 *****
656 *****
657 *****
658 *****
659 *****
660 *****
661 *****
662 *****
663 *****
664 *****
665 *****
666 *****
667 *****
668 *****
669 *****
670 *****
671 *****
672 *****
673 *****
674 *****
675 *****
676 *****
677 *****
678 *****
679 *****
680 *****
681 *****
682 *****
683 *****
684 *****
685 *****
686 *****
687 *****
688 *****
689 *****
690 *****
691 *****
692 *****
693 *****
694 *****
695 *****
696 *****
697 *****
698 *****
699 *****
700 *****
701 *****
702 *****
703 *****
704 *****
705 *****
706 *****
707 *****
708 *****
709 *****
710 *****
711 *****
712 *****
713 *****
714 *****
715 *****
716 *****
717 *****
718 *****
719 *****
720 *****
721 *****
722 *****
723 *****
724 *****
725 *****
726 *****
727 *****
728 *****
729 *****
730 *****
731 *****
732 *****
733 *****
734 *****
735 *****
736 *****
737 *****
738 *****
739 *****
740 *****
741 *****
742 *****
743 *****
744 *****
745 *****
746 *****
747 *****
748 *****
749 *****
750 *****
751 *****
752 *****
753 *****
754 *****
755 *****
756 *****
757 *****
758 *****
759 *****
760 *****
761 *****
762 *****
763 *****
764 *****
765 *****
766 *****
767 *****
768 *****
769 *****
770 *****
771 *****
772 *****
773 *****
774 *****
775 *****
776 *****
777 *****
778 *****
779 *****
780 *****
781 *****
782 *****
783 *****
784 *****
785 *****
786 *****
787 *****
788 *****
789 *****
790 *****
791 *****
792 *****
793 *****
794 *****
795 *****
796 *****
797 *****
798 *****
799 *****
800 *****
801 *****
802 *****
803 *****
804 *****
805 *****
806 *****
807 *****
808 *****
809 *****
810 *****
811 *****
812 *****
813 *****
814 *****
815 *****
816 *****
817 *****
818 *****
819 *****
820 *****
821 *****
822 *****
823 *****
824 *****
825 *****
826 *****
827 *****
828 *****
829 *****
830 *****
831 *****
832 *****
833 *****
834 *****
835 *****
836 *****
837 *****
838 *****
839 *****
840 *****
841 *****
842 *****
843 *****
844 *****
845 *****
846 *****
847 *****
848 *****
849 *****
850 *****
851 *****
852 *****
853 *****
854 *****
855 *****
856 *****
857 *****
858 *****
859 *****
860 *****
861 *****
862 *****
863 *****
864 *****
865 *****
866 *****
867 *****
868 *****
869 *****
870 *****
871 *****
872 *****
873 *****
874 *****
875 *****
876 *****
877 *****
878 *****
879 *****
880 *****
881 *****
882 *****
883 *****
884 *****
885 *****
886 *****
887 *****
888 *****
889 *****
890 *****
891 *****
892 *****
893 *****
894 *****
895 *****
896 *****
897 *****
898 *****
899 *****
900 *****
901 *****
902 *****
903 *****
904 *****
905 *****
906 *****
907 *****
908 *****
909 *****
910 *****
911 *****
912 *****
913 *****
914 *****
915 *****
916 *****
917 *****
918 *****
919 *****
920 *****
921 *****
922 *****
923 *****
924 *****
925 *****
926 *****
927 *****
928 *****
929 *****
930 *****
931 *****
932 *****
933 *****
934 *****
935 *****
936 *****
937 *****
938 *****
939 *****
940 *****
941 *****
942 *****
943 *****
944 *****
945 *****
946 *****
947 *****
948 *****
949 *****
950 *****
951 *****
952 *****
953 *****
954 *****
955 *****
956 *****
957 *****
958 *****
959 *****
960 *****
961 *****
962 *****
963 *****
964 *****
965 *****
966 *****
967 *****
968 *****
969 *****
970 *****
971 *****
972 *****
973 *****
974 *****
975 *****
976 *****
977 *****
978 *****
979 *****
980 *****
981 *****
982 *****
983 *****
984 *****
985 *****
986 *****
987 *****
988 *****
989 *****
990 *****
991 *****
992 *****
993 *****
994 *****
995 *****
996 *****
997 *****
998 *****
999 *****
1000 *****

```

## MBASIC Tokens

By Jeff Richards

THE TOKENS used in Microsoft BASIC have been published many times, but a program that actually finds all the BASIC reserved words, and prints them out with their tokens, may be of interest. The addresses of the routines that deal with each of the tokens are also listed.

This program was developed for Microsoft CP/M BASIC (BASIC-80) Version 5.2, but it should work for most Microsoft BASICs that use the same format.

Two variables may alter with different implementations. First, the value of BASE may change (though obviously for CP/M version it will be 100h). For ROM Basics it will probably be the base of ROM.

Secondly, the value COUNT may have to be adjusted to get the addresses correct. This value is the token of the last keyword that could validly commence a program line. In the case of BASIC 80, this is RESET



# CASH & CARRY software

for IBM P.C. and Columbia M.P.C.

PERFECT WRITER:  
PERFECT CALC:  
PERFECT FILER:  
PERFECT SPELLER:

Revolutionary Word Processor with split screen editing  
Powerful Electronic Spreadsheet with 17 application programs built in  
Easy to use management Data Base with mailing facility  
50,000 word Spelling Dictionary, works in conjunction with Perfect Writer

FUTURE BUSINESS SYSTEMS - Powerful easy to use business package comprising of: Debtor and invoicing, Stock Control, Creditor System, General Ledger. Features Name Key Sort, General Ledger Report Formatter, Utilities, Colour, MS-DOS, Standard Stationery. Please note all features are user prompted and can be stand alone or fully integrated.

CUSTOMIZED TECHNOLOGY PRODUCTS -  
PERTERM: Menu driven communications package, MS-DOS or C/PM 86  
CHOMPS: Eat the monsters before they gobble you up, very popular arcade game  
ISAM: Create your own data base easily with this package  
GAMES PACK II: Ten great games including the old favourite SPACE TREK  
TREASURE HUNT: A 3-D game where you search for the treasure in a mansion.  
CATALOG: A Catalogue system that catalogues your disk and their files  
CHARACTER GENERATOR: This product allows one to easily define (or redefine) graphic characters on an enlarged work tablet  
CROSS REFERENCE: List variables and cross reference basic programs  
ARITHMETIC TUTORIAL: This program takes one through many of the logistics of mathematics from the beginner to advanced level

QUALITY SOFTWARE - GBS DATA BASE, Menu Gen, Report Gen, Quick Gen and GBS System  
Programmable relational database management system. Remarkably Powerful. CP/M 86

CARPE SOFTWARE SYSTEMS - ACCOUNTING PACKAGES includes Debtors, Stock/Invoicing, General Ledger and Creditors Systems. Also includes Couriers, Plumbers, Merchants and Clothing/Shoe trade packages, Bakers Job Costing, Manufacturing and Nursing Home Systems

SAPPHIRE MARS:

Dealers inquiries welcome



One of the most powerful menu driven super financial modelling packages CP/M 86  
SYDNEY: PRESIDENT COMPUTERS 100 GEORGE ST HORNSBY  
MELBOURNE: PRESIDENT COMPUTERS 1/609 ST. KILDA RD MELBOURNE  
BRISBANE: PRESIDENT COMPUTERS 416 LOGAN RD STONES CORNER QLD  
PERTH: PRESIDENT COMPUTERS 91/1 LEURA AVE CLAREMONT W.A.  
ADELAIDE: PRESIDENT COMPUTERS 100 PIRIE STREET ADELAIDE S.A.

PH. (02) 476 2700  
(03) 529 1788  
(07) 397 0888  
(09) 384 5787  
(08) 223 6333





(CCh) with only such keywords as THEN, TO and STEP having higher-valued tokens.

If you want to use this information to do some poking around in the interpreter, have a look at the TRON and TROFF routines for a little bit of very sneaky code.

```

10 DEFINT A-Z
20 BASE=&H100+5
30 COUNT=&HCC
40 C=1
50 FOR I=1 TO 27
60 IF I=27 THEN J=J+1:GOTO 90
70 I1=BASE+256+(I*2)
80 J=PEEK(I1)+PEEK(I1+1)*256
90 IS="":IF I<27 THEN IS=CHR$(I+64)
100 IF PEEK(J)=0 GOTO 240
110 IS=IS+CHR$(PEEK(J)AND &H7F)
120 IF (PEEK(J) AND &H80) GOTO 140
130 J=J+1:GOTO 110
140 K=PEEK(J+1)
150 JS="":IF K<16 THEN JS=""
160 JS=JS+HEX$(K)
170 KS="":IF K>COUNT GOTO 210
180 IF K<64 THEN K=K+COUNT
190 K=(K-128)*2+BASE
200 KS=HEX$(PEEK(K)+(PEEK(K+1)*256))
210 PRINT JS;" ",IS,KS;" "
220 IF INT(C/3)=C/3 THEN PRINT
230 C=C+1:J=J+2:GOTO 90
240 NEXT I

```

# SAMPLE RUN - MICROSOFT BASIC-80 VERSION 5.2

TK	KEYWORD	ADDR	TK	KEYWORD	ADDR	TK	KEYWORD	ADDR
F7	AND		1A	HEX\$	46B1	0E	RETURN	14FA
06	ABS	2866	85	INPUT	188D	0F	REM	1517
0E	ATN	394E	8B	IF	167A	A9	RESUME	15FC
15	ASC	4984	DA	INSTR		CA	RSET	547B
AB	AUTO	1644	05	INT	2A7F	02	RIGHT\$	499B
C3	CLOSE	53DA	10	IMP	280B	08	RND	388A
9A	COMT	4491	FB	IMP		AC	RENUM	22FC
92	CLEAR	4539	DD	INKEY\$		CC	RESET	59BA
1C	CINT	2979	C8	KILL	59D3	BB	RANDOMIZE	248A
1D	CSMG	29F3	88	LET	1541	90	STOP	4435
1E	CDBL	2A1F	B1	LINE	181F	A5	SWAP	44B4
2B	CVI	5180	C4	LOAD	52A1	CB	SAVE	539C
2C	CVS	5183	C9	LBET	547C	D4	SPC(	
2D	CVD	5186	9E	LPRINT	16BD	D1	STEP	
8C	COS	38A6	9F	LLIST	289A	04	SGN	287B
16	CHR\$	4914	1B	LPOS	1D85	07	SQR	36FA
D6	CALL	4D11	93	LIST	289F	09	SIN	38AC
B0	COMMON	1515	0A	LOG	26C2	13	STR\$	46B7
B9	CHAIN	4D94	30	LOC	5644	D8	STRING\$	
04	DATA	1515	12	LEN	48F0	18	SPACES	494D
06	DIM	39A3	01	LEFT\$	496A	BD	SYSTEM	59B3
AD	DEFSTR	13FB	11	LOF	565C	A3	TROW	44AE
AE	DEFINT	13FE	C5	MERGE	5362	A4	TROFF	44AF
AF	DEFSNG	1401	FC	MOD		D0	TAB(	
B0	DEFDL	1404	32	MKID	5167	CE	TO	
98	DEF	1E45	13	MKS\$	516A	CF	THEN	
AA	DELETE	2287	34	MKD\$	516D	0D	TAN	3939
01	END	443A	03	MID\$	49A5	D9	USING	
A2	ELSE	1517	83	NEXT	45C8	D2	USR	
A6	ERASE	44F2	96	NULL	44A5	14	VAL	49C6
A7	EDIT	3D0E	C7	NAME	5064	DC	VARPTR	
A8	ERROR	1639	94	NEW	4354	A1	WIDTH	203C
D6	ERL		D5	NOT		97	WAIT	201C
D7	ERR		9D	OUT	2016	B4	WHILE	4C79
0B	EXP	3757	95	ON	15AB	B5	WEND	4C9C
2F	EOF	5592	BF	OPEN	50B9	B7	WRITE	5001
FA	EQV		F8	OR		F9	XOR	
02	FOR	11EE	19	OCT\$	46AB	F2	+	
C0	FIELD	5418	BA	OPTION	2440	F3	-	
C6	FILES	59FD	C2	PUT	5AF6	F4	*	
D3	FN		99	POKE	22CB	F5	/	
0F	PRE	4B18	91	PRINT	16C5	F6	\	
1F	FIX	2A6C	11	POS	1DEB	FD	^	
09	GOTO	14C3	17	PREK	22C1	DB	'	
09	GO TO	14C3	07	READ	1957	BF	>	
0D	GOSUB	14AB	8A	RUN	1495	F0	=	
C1	GET	5AF7	0C	RESTORE	441A	F1	<	

READ ABOUT

## PRATTL

# DISKS FOR YOUR DGOS

APPLIED TECHNOLOGY S100 CARD OWNERS, NOW THERE IS A STANDARD IBM 8 INCH DISK SYSTEM FOR YOUR COMPUTER.  
(TOTALLY ENDORSED BY APPLIED TECHNOLOGY)

SINGLE SIDED VERSION  
1 x 600K DRIVE

FROM:  
\$1370 + SALES TAX

SINGLE SIDED VERSION  
2 x 600K DRIVE

FROM:  
\$2340 + SALES TAX

DOUBLE SIDED VERSION  
1 x 1.2 MEG DRIVE

FROM:  
\$1804 + SALES TAX

DOUBLE SIDED VERSION  
2 x 1.2 MEG DRIVE

FROM:  
\$2504 + SALES TAX

(CONTROLLER ONLY IS ALSO AVAILABLE)

SYSTEMS INCLUDE CASE, SUPPLY, CABLES, MANUAL, CONTROLLER, DRIVES CP/M 2.2  
MICROSOFT BASIC-80 TOTAL SOFTWARE VALUE NEAR \$600.00 FREE!

ALSO A BROAD RANGE OF OTHER S100 CARDS TO SUIT YOUR SYSTEM.



## MICROCOMPUTER PRODUCTS

ACOUSTIC ELECTRONIC DEVELOPMENTS PTY., LTD. INC. IN N.S.W.

9-5  
6 DAYS/WEEK  
PHONE: 681-4966  
TELEX: AA70664



OFFICIAL SYSTEM SUPPORT ORGANISATION FOR APPLIED TECH. S100 SYSTEMS

# SHARP (AND TANDY) PCs

## Spaceship Lander

By C. Colle

WITH Spaceship Lander, you can simulate the landing of a spaceship on any planet in the solar system. The table gives you the surface gravity values for the planets and some of their satellites.

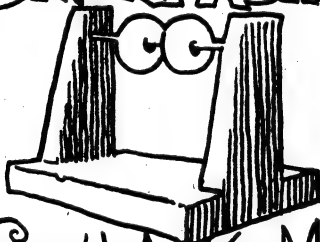
The gravity used in the sample run (1.62) is for Earth's moon, and is a good landing (soft landing and less than 200 metres from target). To have a successful landing, your speed must be less than or equal to eight metres per second when you're four metres or less above the ground.

To succeed, you have to input two accelerations, one to reduce the speed, the other to get nearer the target. If you slow down too much, the speed becomes negative, meaning that you're going up instead of down. The height, too, increases instead of decreasing. You can input an acceleration of zero, meaning free fall.

Surface Gravity Values (in  $m/s^2$ )

Mercury	3.95	Callisto	3.20
Venus	8.72	Saturn	8.77
Earth	9.81	Uranus	9.46
Moon	1.62	Neptune	13.66
Mars	3.84	Pluto	4.85
Jupiter	23.16	Ceres	0.85
Ganymede	3.43	Palas	0.54
Io	2.26	Juno	0.21
Europa	1.58	Vesta	0.43

**EXCLUSIVE!**  
FOR VG READERS  
BINARY ABACUS



SEND US ALL YOUR MONEY  
& GET ONE FREE!!!

```

10:"L"PRINT "SP
    ACESHIP LAND
    ER"
15:S=1.3:D=5:M=
    2000:P=750
20:BEEP 1:INPUT
    "GRAVITY=" ;
    G
30:F=4*G*M:A=((
    4*F)/(3*M))-
    G:V=54*(F/M)
    :U=0:Y=(V^2)
    /(2*A):X=V
40:J=((2*Y+V^2/
    G)/(1+A/G)):
    I=(J(J/A)*P)
    /(F*S)
50:Q=M-P
55:H=INT (Y):K=
    INT (Y):L=
    INT (Q):N=
    INT (X)
60:PRINT "HEIGH
    T=";H;"SPEED
    =" ;K
70:PRINT "FUEL=
    ";L;"DISTANC
    E=";N
80:IF M=P THEN 1
    50
90:BEEP 1:INPUT
    "VERT.ACCELE
    RATION=" ;I;
    "HORIZ.ACCEL
    ERATION=" ;W
100:A=T*F/E2
110:B=W*F/E2
120:M=M-(((A+B)*
    I*D)/500)
130:IF M>P THEN 1
    50
140:A=0:B=0:M=P
150:Z=(D+(G-A/M)
    *6)
160:IF Z<0 LET Z=
    Z*-1
170:IF A=0 LET V=
    V+Z:GOTO 190
180:V=V-Z
190:U=U+D*B/M
200:Z=V*D*1.2
220:Y=Y-Z
230:Z=U*D
250:X=X-Z
260:IF Y<=4 THEN
    280
270:GOTO 50
280:IF V<=8 THEN
    310
285:GOSUB 500
290:PRINT "YOU C
    RASHED AT " ;
    K;"M/S"
300:PRINT N;" M.
    FROM TARGET"
    :GOTO 380
310:IF X>200 THEN
    350
320:GOSUB 500
330:PRINT "CONGR
    ATULATIONS.
    YOU"
340:PRINT "ARE "
    ;N;" M. FROM
    TARGET":
    GOTO 380
350:GOSUB 500
360:PRINT "GOOD
    LANDING BUT"
370:PRINT N;" M.
    FROM TARGET"
380:INPUT "ANOTH
    ER GAME ? " ;
    W$
390:IF W$="YES"
    THEN 15
400:END
500:N=ABS (INT (
    X))
510:RETURN

GRAVITY= 1.62
HEIGHT= 12250.
    SPEED= 414.
FUEL= 1250.
    DISTANCE= 414.
VERT. ACCEL.=5.
HORIZ. ACCEL.=5.
HEIGHT= 9994.
    SPEED= 376.
FUEL= 1215.
    DISTANCE= 406.
VERT. ACCEL.=5.
HORIZ. ACCEL.=5.

```

```

HEIGHT= 7969.
    SPEED= 337.
FUEL= 1181.
    DISTANCE= 389.
VERT. ACCEL.=0.
HORIZ. ACCEL.=5.
HEIGHT= 5652.
    SPEED= 386.
FUEL= 1164.
    DISTANCE= 364.
VERT. ACCEL.=0.
HORIZ. ACCEL.=5.
HEIGHT= 3044.
    SPEED= 434.
FUEL= 1147.
    DISTANCE= 331.
VERT. ACCEL.=100
HORIZ. ACCEL.
HEIGHT= 1660.
    SPEED= 230.
FUEL= 789.
    DISTANCE= 287.
VERT. ACCEL.=50.
HORIZ. ACCEL.=5.
HEIGHT= 846.
    SPEED= 135.
FUEL= 602.
    DISTANCE= 230.
VERT. ACCEL.=50.
HORIZ. ACCEL.=5.
HEIGHT= 742.
    SPEED= 17.
FUEL= 414.
    DISTANCE= 160.
VERT. ACCEL.=0.
HORIZ. ACCEL.=0.
HEIGHT= 346.
    SPEED= 65.
FUEL= 414.
    DISTANCE= 90
VERT. ACCEL.=10.
HORIZ. ACCEL.=0.
HEIGHT= 35.
    SPEED= 51.
FUEL= 380.
    DISTANCE= 20.
VERT. ACCEL.=25.
HORIZ. ACCEL.=0.
CONGRATULATIONS.
YOU ARE 50.METRE
S.FROM TARGET

```

READ ABOUT

**PRATTL**



# Mark Prediction

By C. Colle

THE MARK Prediction program will be particularly useful to students wanting to know, in advance, their average grade marks for the year.

Enter all the marks or ratings in any subject that you have received, and then those which you would like to get, and the program will tell you the percentage chance of this. However, you must assume that your work is constant. You can also find out the mean and standard deviation.

X is the mark to be entered, and Y is the frequency of this mark. When all marks have been entered, type O for the last X, then enter the mark you would like to get.

The program listing is for the printer. Press ENTER for your chances for a different mark. However, if this mark is for another subject, press SHFT M, as the memories must be cleared first.

```
10:"M"PRINT "MARK PREDICTION"
N":CLEAR
13:INPUT "ENTER X ";X
15:PRINT "X = "
:Y
20:IF X=0THEN 8
O
27:INPUT "ENTER Y ";Y
30:PRINT "Y = "
:Y
40:A=YX:B=B+Y:C=C+A:D=XX:E=E+Y:F=F+E:G=C
C
70:GOTO 13
80:M=C/B
90:PRINT "MEAN = ";USING "#.###";M
100:S=J((F-(G/B))/B)
105:Z=S/J(1-1/B)
110:PRINT "STANDARD DEVIATION="
:Z
```

```
120:V=ZZ
130:Q=0:P=0:R=0:
R=0:T=0
137:INPUT "DESIRE MARK ? ";N
140:PRINT " DESIRE MARK ? "
:IN
150:Q=N:GOSUB 20
O
160:Q=4R:Q=N-.5:GOSUB 200
170:Q=Q+R:Q=N+.5:GOSUB 200
180:Q=Q+R:P=6.65
Q/Z:BEEP 2
190:PRINT "YOU HAVE AVE ";P;" % CHANCES":GOTO 130
200:T=(Q-M)*(Q-M)/V:R=J(EXP -T):RETURN
```

```
X = 7.
Y = 1.
X = 6.
Y = 3.
X = 5.
Y = 1.
X = 0.
MEAN = 6.0
STAND. DEVIATION = 0.7
DESIRED MARK ? 6.0
YOU HAVE 52.2 % CHANCES
DESIRED MARK ? 7.0
YOU HAVE 22.1 % CHANCES
DESIRED MARK ? 5.0
YOU HAVE 22.1 % CHANCES
```

MARK PREDICTION  
\*\* SAMPLE RUN \*\*

# Alien Attack

By Scott Story

ALIEN ATTACK starts with the Alien moving along the screen to base. The attack can be checked and stopped, after five hits, by pressing the 7 button (which stops the invader when it appears on the top of the screen), the 4 button (which stops the invader in the middle of the screen) or the 1 button (which stops the invader at the bottom of the screen).

The alien then begins another attack, each time moving faster and faster as it approaches the base. You score a point each time you hit the alien.

```
1:P=1:F=0:B=40
10:"A":CLEAR:WAIT 100:PRINT "ALIEN ATTACK START"
20:DIM T$(3)
30:T$(1)="0402030204"
40:T$(2)="10080C0810"
50:T$(3)="4020302040"
60:GDCURSOR 110
75:CLS:P=1:F=F+1:S=0:B=B-1
80:FOR I=1TO 11
```

```
90:GOSUB 5005
100:A$=INKEY$
110:IF A$="7"AND R=1GOTO 8000
:1GOTO 8000
120:IF A$="4"AND R=2GOTO 8000
:2GOTO 8000
130:IF A$="1"AND R=3GOTO 8000
:3GOTO 8000
140:P=P+10
150:NEXT I
155:L=F*S+S-5
165:GOTO 8300
200:FOR A=0TO 20
210:BEEP 1,A
220:NEXT A
225:GDCURSOR 1
230:PAUSE "ATTACK HALTED ":GOTO 300
300:CLS:P=1:F=F+1:S=0:B=B-2
310:GOTO 80
5005:CLS
5007:WAIT 10:GDCURSOR 120:GPRINT "7F7F7F"
5010:R=RND(3)
5020:GDCURSOR P
5030:WAIT B:GPRINT T$(R):BEEP 1,60
5050:RETURN
```

```
8000:S=S+1
8005:GDCURSOR P
8007:BEEP 3,50,50:BEEP 3,150,100
8010:WAIT 100:GPRINT "081C3E1C08"
8025:IF S=5GOTO 200
8030:GOTO 140
8300:GDCURSOR 120
8310:GPRINT "7F7F7F"
8400:CLS
8405:GDCURSOR 130
8420:GDCURSOR 105
8430:GPRINT T$(R)
8450:GDCURSOR 111
8455:WAIT 25:GPRINT "0808"
8460:GDCURSOR 111
8465:WAIT 25:GPRINT "0808"
8470:GDCURSOR 111
8475:WAIT 25:GPRINT "0808"
8480:GDCURSOR 111
```

```
8485:WAIT 25:GPRINT "0808080808080808"
8490:GDCURSOR 111
8495:WAIT 25:GPRINT "0808080808080808"
8500:GDCURSOR 125
8510:WAIT 55:GPRINT "24151E3E7F"
8515:GDCURSOR 125
8520:WAIT 60:GPRINT "14492C126C7F"
8525:GDCURSOR 125
8530:WAIT 60:GPRINT "0A201C282A"
8600:PAUSE "YOUR SCORE-";L
9000:BEEP 1,90,50:BEEP 1,70,50
9010:BEEP 1,150,90:BEEP 1,150,100
9020:BEEP 1,50,60:BEEP 1,250,150
9030:END
```

# COMMODORE

## Ship Maths

By Ric Kube

I DEvised the Space Maths program primarily for my son to learn his addition, but it can be changed to work any of the four functions + - \* and / by changing line 30.

The program gives 10 problems (minimum) and keeps score. If you get five wrong, a red alert sounds and the ship travelling across the top sinks and a new game begins. The ships and waves are made using the programmable characters. This is found in lines 30-160.

Variables:

A\$-F\$ - different parts of the ships.

G\$ - waves on sea

H\$-R\$ - erases unwanted strings to produce movement

S and CT - top left corner of screen for pinking and colour

H - loop for giving problems and moving ship

SC - score

W - sums wrong

A - loop for ocean.

Some of the lines are a little long and I needed to abbreviate as much as possible.

```

5 REM**FOR VIC 20**SEE NOTES AT END
10 REM**SHIP MATHS BY RIC KUBE WALKERIE SA 5330 (C) 1982
20 PRINT"(clear)(black)WHAT IS YOUR NAME? INPUT N"
30 PRINT"(clear): POKE36869,255: SC=7680: CT=30400
40 DATA 7168,0,0,0,0,13,13,13,13,0,0,0,128,128,128,128
50 DATA 7296,13,13,13,13,127,63,31,15,128,248,248,248,255,255,255
60 DATA 7184,129,195,183,127,127,63,63,31,0,128,16,56,248,224,224,248
70 DATA 7312,15,7,3,1,0,0,0,248,248,248,248,248,124,62,30
80 DATA 7280,0,0,0,0,0,1,3,0,0,0,64,192,255
90 DATA 7328,3,3,3,3,3,3,255,255,248,248,248,248,255,255
120 DATA 7216,1,3,7,15,31,63,127,255,1,3,7,15,31,63,127,255
130 DATA 7344,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255
140 DATA 0
150 READ N: IFN THEN FOR N=M TO N+5: READ A: POKE N,A: NEXT: GOTO 150
160 FOR I=7424 TO 7431: POKE I,0: NEXT
170 PRINT"(clear)"
180 A$="(down)(black)A(down)(2 left)P"
190 B$="(down)(black)B(down)(2 left)R"
200 C$="(2 down)(black)C"
210 D$="(down)(black)D(down)(2 left)T"
220 E$="(2 down)(black)E"
230 F$="(2 down)(black)F"
240 G$="(cyan)(3 down)G(down)(2 left)V"
250 H$="(home)(9 down)(7 right)(9 spaces){11 left}{11 spaces}"
260 G$="(down)(2 spaces)(down)(2 left)(2 spaces)"
270 R$="(2 down)(2 spaces)"
280 PRINT"(home)": TAB(8):A$
290 FORA=0 TO 20: PRINT TAB(A):R$NEXT
300 PRINT"(home)(9 down)(7 right)(rev)SHIP MATHS(down)(11 left)(green)
(rev)ADD THEM UP"
310 POKE36877,180: FOR L=1 TO5: D=INT(RND(1)*5)*50+50: FOR M=37015:
POKE36878,M
320 FOR N=1700: NEXT N,M: FOR M=15015 STEP-1: POKE36879,M: FOR N=
1700: NEXT N,M,L
330 POKE36878,0: POKE36877,0
340 FOR A=0 TO21: POKE36877,A*22,250: POKE36877,A*22,3: NEXT
350 PRINT"(home)(right)(green)(19 down)(rev)"N$
360 FOR H=0 TO19 STEP2: PRINT"(home)":TAB(H):A$
370 PRINT H$
380 X=INT(RND(1)*12): Y=INT(RND(1)*12): Z=X+Y
390 PRINT"(home)(20 down)(rev)(blue)SCORE"SC
400 PRINT"(home)(20 down)(rev)(red)(14 right)WRONG"W
410 PRINT"(home){11 down}(22 spaces)"
420 PRINT"(home){11 down}(6 right)(rev)(blue)"X*Y="": INPUTU
430 IF U=2 THEN 450
440 IF U<X<Y THEN 480
450 SC=SC+1: PRINT"(home)":TAB(H):CT: NEXT
460 PRINT"(clear): POKE36878,15: FOR I=128 TO255: POKE36879,I:
POKE36876,I: NEXT: POKE36876,0: POKE36876,0
470 POKE36879,27: GOTO510
480 PRINT"(home)(rev)(7 down)(3 right)DID IT WAS?": W$="": IFW$
THEN 560
490 PRINT"(home)(7 down)(22 spaces)"
500 GOTO 380
510 PRINT"(home)(7 down)(rev)(y=1)YOU GOT A SCORE OF"SC: PRINT"(5 right)
(rev)(red)WELL"WRONG"
520 PRINT"(home){11 down}(6 right)(rev)(mur)ANOTHER GO?" : PRINT"(down)
(9 right)(rev)(Y/N)"
530 GET A$: IF A$="" THEN 530
540 IF A$<>"Y" THEN POKE36869,240: POKE36879,27: PRINT"(clear)(blue)":END
550 SC=0: W=0: GOTO 160
560 POKE36878,15: FOR L=1 TO10: FOR M=1807035 STEP2: POKE36876,M:
FOR N=17010: NEXTN,M
570 POKE36876,0: FOR M=170100: NEXTM,L: POKE36878,0
580 PRINT"(home)":TAB(H):B$: FOR A=170400: NEXT: GOSUB 640
590 PRINT"(home)":TAB(H):C$: FOR A=170400: NEXT: GOSUB 640
600 PRINT"(home)":TAB(H):D$: FOR A=170500: NEXT: GOSUB 640
610 PRINT"(home)":TAB(H):E$: FOR A=170600: NEXT: GOSUB 640
620 PRINT"(home)":TAB(H):F$: FOR A=170700: NEXT: GOSUB 640
630 GOTO 510
640 PRINT"(home)":TAB(H):G$: RETURN
650 PRINT"(home)":TAB(H):R$: RETURN
700 REM ** TO FIT SOME LINES YOU NEED TO ABBREVIATE KEYWORDS **
710 REM ** SHIP MATHS ** (c) RIC KUBE WALKERIE 5330 (085) 412375
720 REM ** IF USING EXPANDED VIC CHANGE VALUE OF 'S' AND 'CT' IN 30**
730 REM ** TO 4096 AND 37888 **

```

READ ABOUT  
**PRATTL**

# G-PASCAL

## The Professional Development System for \*Games \*Utilities \*Commercial software

G-Pascal is probably the most useful program you will buy for your Apple! Its user friendliness and extremely fast compile speed (between 4,000 and 6,000 lines per minute) make it a joy to use for professionals and beginners alike. Its simplicity in fact makes it ideal for schools. The built-in text editor is very easy to use — you can actually use the Editor on its own for simple word processing applications. Operation of the compiler is very straightforward — error messages are in plain English, not numeric codes. Also available is a runtime system which will run programs independently of the compiler.

Ask your local dealer, or send cheque, money order or Bankcard number direct to us.  
All orders shipped by airmail at no extra charge.

### FEATURES:

- \* Comprehensive Pascal subset
- \* Includes compiler, editor and interpreter
- \* Will run on 48K or 64K Apple with DOS 3.3
- \* Support for printers and 80 column cards
- \* Complete system only occupies 14K of memory
- \* Includes Trace and Debug facility
- \* Supports graphics, music, assembler calls
- \* Powerful text editor
- \* Compiles at between 4,000 and 6,000 lines per minute

G-Pascal compiler	\$85.00
Runtime system	\$35.00
Adventure game	\$35.00*
Missile Attack game (* G-Pascal source supplied)	\$35.00*
Starter Kit	
(Compiler, Runtime system plus one game)	\$110.00

Please write or phone for more information.  
P.O. Box 124, Ivanhoe, Victoria, Australia 3079.  
Telephone: (03) 497 1283

**Gammon & Gobbett**  
**Computer Services Pty Ltd**



## Symmetry

By Stewart Collins

SYMMETRY is a program that produces symmetrical patterns on a screen, and can only be stopped by pressing the space bar. It fits in one kilobyte of memory.

```

LET N=30
LET M=21
LET U=0
LET D=-1
30 LET X=RND*M-M+RND
40 LET Y=D-RND*X+RND
50 FOR R=M TO N
60 PLOT N+X,M+Y
70 LET T=X
LET S=U
90 LET D=D-2*(D AND S)
100 LET U=S+D
110 LET X=X*U+Y*S
120 LET Y=T*S-Y*U
130 NEXT A
140 GOTO N

```

Execute by typing RAND and GOTO 0. ☐

## Catcher

By N. Weaver

THE IDEA of Catcher is to catch as many Os as possible with the ). The program uses the RND function to set the height of the "ball" and loops to move it across the screen. The INKEY\$ function controls the height. The main part of the program is in lines 17 to 27. Line 33 decides whether you have the top score.

```

1. LET Z = 0
3. LET A$ = ""
5. PAUSE 100
7. LET A = 0
9. LET C = 14
11. CLS
13. FOR J = 1 TO 10
15. LET B = INT (RND X 19)
17. FOR I = 0 TO 15
19. PRINT AT B,I;"O";AT C,15;"")
21. IF C = B AND I = 15 THEN LET A = A+1
23. LET C = C + (INKEY $ = "6") - (INKEY $ = "7")
25. CLS
27. NEXT I
29. NEXT J
31. PRINT AT 10, 0; "SCORE =";A
33. IF A > Z THEN GOTO 39

```

```

35. PRINT AT 14,0;" TOP SCORE="; Z;" BY "; A$
37. GOTO 5
39. PRINT AT 11,0; "TOP SCORE. NAME ? ";
41. INPUT A$
43. PRINT A$; " "; A
45. LET Z = A DOWN-6
47. RAND Z UP-7
49. GOTO VAL "3"

```

## Resistors

By Alan Hill

RESISTORS is a simple but useful program which runs on a Sinclair ZX80 with eight kilobytes of ROM and four kilobytes of RAM.

It should also run in one kilobyte, but not with the old ROM because of the arithmetic limitations.

```

10 PRINT "THIS PROGRAM CALCULATES
OVERALL RESISTANCE."
20 PRINT
30 PRINT "ARE RESISTORS IN SERIES ? (S)"
40 PRINT "OR IN PARALLEL ? (P)"
50 INPUT A$
60 IF A$="S" THEN GOTO 100
70 IF A$="P" THEN GOTO 400
80 GOTO 30
100 CLS
110 PRINT "HOW MANY RESISTORS ARE?"
120 PRINT "IN SERIES TOGETHER ? ";
130 INPUT N
140 PRINT N
150 PRINT

```

## Typing Tutor

By Peter McKay

THE AIM of Typing Tutor is to teach the user to type by being given lines five words long, to be typed as fast as possible. At the end of each line, the time is shown in seconds. Before the main program will run, program two must be run with data for the first sentences.

Sample data:

(R\$) - Jana, Rhys, Thom, Jane, Jack Scott.  
(Z\$) - Threw, Found, Passed, Pushed, Helped, Sat on.  
(S\$) - Fat, Big, Cold, Sick, Smug, Huge.  
(O\$) - Hog, Pig, Cow, Cat, Rat, Dog.

Words of nearly the same length must be used so that spaces between the words are few. The program displays the sentence, and the last letter pushed appears in inverse. ☐

```

1000 DIM A$(6,4)
1100 DIM Z$(6,6)
1200 DIM S$(6,4)
1300 DIM O$(6,3)
1400 FOR I=1 TO 6
1500 INPUT A$(I)
1600 INPUT Z$(I)
1700 INPUT S$(I)
1800 INPUT O$(I)
1900 NEXT I

```

```

1 REM "TYPING TUTOR"
2 LET R=INT (RND*6)+1
3 LET Z=INT (RND*6)+1
4 LET S=INT (RND*6)+1
5 LET O=INT (RND*6)+1
10 LET M$=R$(R)+" "+Z$(Z)+" TH
E "+S$(S)+" "+O$(O)
15 PRINT AT 0,1,M$
20 FOR I=1 TO LEN M$
21 IF INKEY$="" THEN GOTO 21
22 LET A$=INKEY$
23 IF A$=CHR$ 118 THEN LET A$=
24 LET W=I
25 IF A$<>M$(I TO I) THEN GOTO
21
26 IF I=1 THEN GOTO 111
27 IF W=1 THEN LET W=2
28 PRINT AT 0,W-1,M$(W-1 TO W-
1)
30 PRINT AT 0,I,CHR$ (CODE M$(
I TO I)+128)
40 NEXT I
50 LET T=INT (65535-PEEK 16436
-256*PEEK 16437)/50
55 PRINT AT 1,0;"SCORE="
60 PRINT AT 1,0;"SCORE=";T
100 GOTO 1
111 POKE 16437,255
112 POKE 16436,255
113 GOTO 27
150 SAVE "TYPING TUTOR"
160 GOTO 1

```

```

170 PRINT "INPUT VALUES IN OHMS"
190 LET R=0
200 DIM V(N)
230 FOR J=1 TO N
240 PRINT "R";J;" ? ";
250 INPUT V(J)
260 PRINT V(J)
270 LET R=R+V(J)
280 NEXT J
290 PRINT
300 PRINT "R = ";R;" OHMS"
310 PRINT AT 20,0;"HIT NEWLINE
FOR NEXT CALCULATION"
320 INPUT Z$
330 CLS
340 GOTO 10
400 CLS
410 PRINT "HOW MANY RESISTORS
ARE"
420 PRINT "IN PARALLEL
TOGETHER ? ";
430 INPUT N
440 PRINT N
450 PRINT
460 LET RR=0
470 DIM A(N)
480 PRINT "INPUT EACH RESIST
ANCE IN OHMS"
490 FOR K= 1 TO N
510 PRINT "R";K;" ? ";
520 INPUT A(K)
530 PRINT A(K)
540 LET RR=RR+(1/A(K))
550 NEXT K
560 PRINT
570 PRINT "R = ";1/RR;" OHMS"
580 GOTO 310

```

## Australia

By K.E. Johnstone

AS A TEACHER using a Sinclair ZX81 in primary grades, I've found that most programs seem to be aimed at mathematics or spelling. However, here's a truly Aussie program which has proved popular with children.

A map of Australia is printed and the user is asked to name each state. When all the states are named correctly, the name is done for the capital cities.

The program is entered in two parts – the first part sets up the map:

```

5 REM "AUSTRALIA"
10 DIM A$(132,2)
20 PRINT 1
30 FOR N=1 TO 132
40 INPUT A$(N)
50 CLS
60 PRINT N; AT 0,4;A$(N); AT 2,0;N+1
70 NEXT N
80 CLS

```

At this stage, the program should be RUN and the 132 pairs of co-ordinates entered as follows:

```

MV MU NT NS OR PR PQ PP QO QN RM SL TK UJ VJ VH WG XF XE XD
XC YB XA X9 W8 V7 V6 V5 U4 U3 T2 S2 R2 Q1 P2 D1 N1 M2 L2 K3
K4 J5 J6 I7 H6 H5 G5 G6 F7 E7 D7 C8 B8 A7 96 86 76 65 54 43
33 22 12 01 .1 .2 .3 .4 .5 .6 .7 .8 *9 -A *B -C -D -E -F *G
/H .I .J .K .L .M .N .O .P .Q .R .S .T .U .V .W .X .Y .Z
9T AT BU CT DT ET FU GY GT FS FR GP GQ HP IO JO KO LP LQ LR
LS LT R, R; R/O * P* Q/ O; O; P; O;

```

Lines 10 to 80 may now be deleted. Remember that the map co-ordinates are now stored as A\$ so don't use RUN.

```

160 LET Z=Z+100
165 PAUSE 300
170 CLS
175 IF Z=700 THEN GOTO 90
180 PRINT TAB 3;"NAME THE CAPITAL."
190 GOTO 105
200 FOR N=1 TO 132
210 PLOT CODE A$(N,1)-19, CODE A$(N,2)-19
220 NEXT N
250 FOR N=17 TO 39 STEP 2
260 PLOT 19,N
270 NEXT N
280 FOR N=19 TO 31 STEP 2
290 PLOT N,25
300 NEXT N
310 FOR N=25 TO 13 STEP -2
320 PLOT 31,N
330 NEXT N
340 FOR N=32 TO 40 STEP 2
350 PLOT N,22
360 NEXT N
370 PLOT 41,21
371 PLOT 42,22
380 PLOT 32,16
381 PLOT 34,15
383 PLOT 36,15
384 PLOT 38,14
390 FOR N=25 TO 33 STEP 2
400 PLOT 29,N
410 NEXT N
420 PRINT AT 13,18;" "
430 RETURN
440 REM 13 ,18 = AUSTRALIAN CAPITAL TERRITORY BUT NAME TOO LONG
FOR SCREEN
510 LET B$="QUEENSLAND"
515 GOTO 1000
520 LET B$="(NEW SOUTH WALES"
525 GOTO 1000
530 LET B$="(VICTORIA"
535 GOTO 1000
540 LET B$="(TASMANIA"
545 GOTO 1000
550 LET B$="(NORTHERN TERRITORY"
555 GOTO 1000
560 LET B$="(SOUTH AUSTRALIA"
565 GOTO 1000
570 LET B$="(WESTERN AUSTRALIA"
575 GOTO 1000
610 LET B$="(DARWIN"
615 GOTO 1000
620 LET B$="(BRISBANE"
625 GOTO 1000
630 LET B$="(SYDNEY"
635 GOTO 1000
640 LET B$="(MELBOURNE"
645 GOTO 1000
650 LET B$="(HOBART"
655 GOTO 1000
660 LET B$="(ADELAIDE"
665 GOTO 1000
670 LET B$="(PERTH"
1000 FOR M=3 TO LEN B$
1010 IF B$(M)="/" THEN PRINT AT CODE B$(1),CODE B$(2)+M-3;"?";
1020 IF B$(M)="" THEN PRINT " ";
1030 NEXT M
1100 INPUT C$
1110 IF C$=B$(3 TO) THEN GOTO 1200
1120 PRINT AT CODE B$(1),CODE B$(2);C$
1130 LET Q(R)=1
1140 GOTO 150
1200 PRINT AT 21,0;"WRONG"
1205 REM. PAUSE MAY BE USED FOR SLOWER READERS
1210 FOR N=370 TO LEN B$
1220 PRINT AT CODE B$(1),CODE B$(2)+N-3;CHR$(CODE B$(N)+128)
1230 NEXT N
1240 PRINT AT 21,0;" "

```

```

1250 LET F=F-1
2000 GOTO 150
9990 SAVE "AUSTRALIA"
9999 GOTO 90
Note line 620 - B$(1) is Graphics A.

```

## Chaser

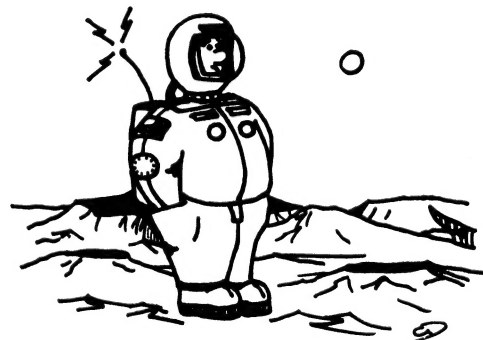
By N. Weaver

TRYING TO avoid being eaten by a monster is the aim of Chaser. When running the program, GET READY will appear, followed by a pause. Three dots will then appear on the screen, one with an H which is home, one on the left which is the monster, and the remaining dot on the right – you. The main part of the program is in lines 31 to 51.

```

1. PRINT "GET READY" (INVERSE)
3. LET Z = 1
5. LET A = INT (RND × 200)
7. LET C = INT (RND × 64)
9. PAUSE A
11. LET B = INT (RND × 44)
13. CLS
15. LET X = 63
17. LET Y = 30
19. LET L = 0
21. LET M = 30
23. PLOT X,Y
25. PLOT L,M
27. IF X = L AND M = Y THEN GOTO 57
29. PLOT C,B
30. PRINT "H"
31. IF INKEY$="H" THEN LET X = X + Z
33. IF INKEY$="G" THEN LET X = X - Z
35. IF INKEY$="Y" THEN LET Y = Y + Z
37. IF INKEY$="B" THEN LET Y = Y - Z
39. IF X = C AND Y = B THEN GOTO 65
41. IF INKEY$="Q" THEN LET Z = 2
43. IF INKEY$="Q" THEN LET Z = 1
45. IF L < X THEN LET L = L + 1
47. IF L > X THEN LET L = L - 1
49. IF M < Y THEN LET M = M + 1
51. IF M > Y THEN LET M = M - 1
53. CLS
55. GOTO 15
57. CLS
59. PRINT "YOU'RE DEAD"
61. GOTO 69
63. STOP
65. CLS
67. PRINT "YOU WIN"
69. RUN

```



"BREAKER, BREAKER, THIS HERE'S THE RUBBER DUCK - YOU GOT YOUR FARS ON, C'MON?"